BUON ANNO

MSX COMPUTER MAGAZINE

N. 6 GEN./FEB. 1986 Sped in abb. post. Gr. III L. 9.000

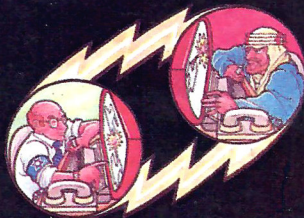
ECCEZIONALE NOVITÀ

IN TUTTE LE EDICOLE

TUTTO QUEL CHE SERVE

PER ENTRARE NEL FAVOLOSO MONDO DELLA COMPUTER-COMUNICAZIONE VIA TELEFONO

Hai un computer e un telefono? Questa rivista è fatta apposta per te! Tutto l'hardware e il software (su cassetta!) per comunicare via computer e telefono in Italia e in tutto il mondo. Potrai metterti in contatto con altri computer, trasmettere e ricevere posta, catturare programmi, trovare i mille amici (gli hackers!) del Modem Club International con cui scambiare programmi e informazioni... più una banca dati diretta a disposizione 24 ore... più altro ancora...

PIÙ I NUMERI DI TELEFONO PER PORSI IN CONTATTO

CON IL FASCICOLO, GRATIS IL SOFTWARE SU CASSETTA

MSX Computer Magazine è edita da Arcadia srl,
C.so Vitt. Emanuele 15, Milano.
Tel. 02/797830 (solo giovedì h. 15-18).
Una copia L. 9.000.
Fotocomposizione: Composit.
Stampa: Garzanti,
Milano. Distribuzione: SO.DI.P. Angelo
Patuzzi srl, Via Zuretti 25, Milano.
Registrato Trib. Milano N. 52 del 2/2/85.
Resp. Sira Rocchi.
Sped. in abb. post. Gr. III/70.
MSX is a trademark of MicroSoft Co.
Manoscritti, disegni, fotografie
e programmi inviati non si
restituiscono anche se non pubblicati.

IN QUESTO NUMERO

BUON ANNO!

A TUTTI IN REGALO
IL CALENDARIO 1986

GLI ARTICOLI

- MSX TAPE SOFT
- BASIC DIDATTICA
- LINGUAGGIO MACCHINA
- SCROLL WINDOW

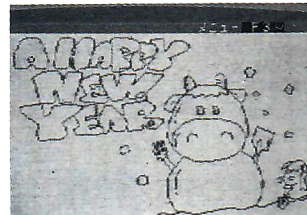
I PROGRAMMI

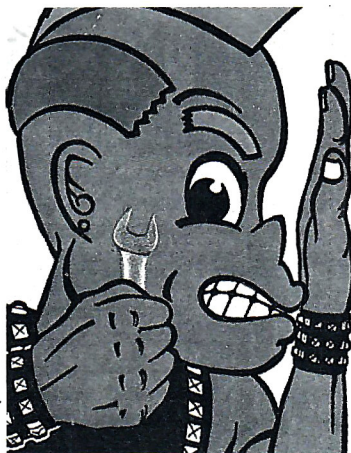
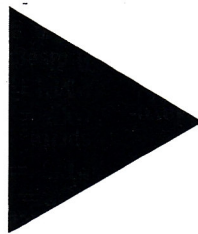
- L'IDRAULICO
- AMERICAN FOOTBALL
- IPPICA
- ROTATE
- PLOTTER
- UFO
- CODICE STRADALE
- MUSICA
- SPRITES GRANDI
- ARCHIVIO LIBRI

Cover Msx Tokyo courtesy.

MSX TAPE SOFT

I programmi che vi presentiamo in questo sesto numero di MSX Computer Magazine sono tutti compatibili con qualsiasi sistema MSX. Ecco per voi ben 10 programmi.

Ricordate di collegare la spina del controllo motore alla presa REM del vostro registratore, se quest'ultimo la possiede. Assicuratevi che la spina rossa sia collegata alla presa MIC del

registratore e la spina nera alla presa EAR. Se il vostro mangiacassette non possiede la presa REM, fate particolare attenzione a quando un programma è stato caricato o deve essere caricato, affinché il nastro scorra per il giusto tempo. Appena vedete apparire sul video, dopo un comando di caricamento, la scritta OK, spegnete

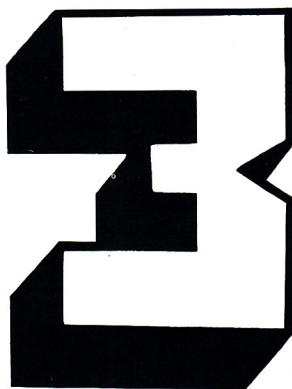
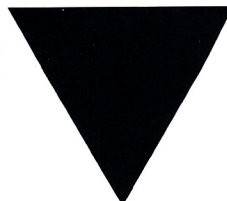
1 L'IDRAULICO
("IDRA")
adatt. da M. Zanchi

Mario l'idraulico ha dei problemi: l'acquedotto municipale perde ed il suo compito è di controllare centimetro per centimetro tutte le tubazioni, dovete quindi aiutarlo facendo in modo che non cada dai grossi tubi. Per passare al quadro successivo dovete raggiungere il barile in cui si riversa l'acqua prima dello scadere del tempo. Quest'ultimo è scandito da una striscia che si cancella a poco a poco posta sul lato destro dello schermo. Quando il tempo finirà, Mario perderà una vita e dovrete ricomincaire il percorso da capo.

Le perdite delle tubazioni sono causate da uno strano e pericoloso scorpione meccanico che, oltre a mangiare le giunture dei tubi, cercherà di eliminarvi con il suo morso letale. Per difendersi dallo scorpione bisogna colpirlo con un grosso martello. Se si passa sopra un foro in un tubo, si perderà del tempo per operare una riparazione rischiando così di non riuscire a terminare in tempo l'intero percorso.

Per spostare Mario si può usare il joystick o i tasti cursore. Per lanciare il martello si usa o il tasto di spazio oppure, se si usa il joystick, il tasto fire. Se si vuole giocare con la tastiera, far partire il gioco premendo il tasto di spazio altrimenti il tasto fire.

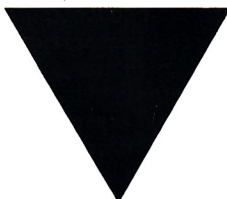
Per chi volesse analizzare il listato, interrompere il programma con CTRL+STOP e digitare SCREEN 0 + ENTER.

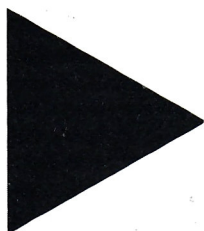
il registratore.
ATTENZIONE: la presentazione va caricata con il comando:
BLOAD "MSXCMP", R
tutti gli altri programmi, invece, con l'istruzione CLOAD (i caratteri racchiusi tra virgolette, nelle parentesi, rappresentano il nome con cui è stato registrato il programma).
Nei riquadri, i commenti relativi ai

vari giochi e utility.
Nella cassetta allegata a questo fascicolo troverete sul lato A: Presentazione, L'idraulico, American Football, Ippica, Rotate, Plotter.
Sul lato B:
UFO, Codice stradale, Musica, Sprites grandi, Archivio libri.

MUSICA
("MUSICA")

di M. Fontana

Nella stesura dei programmi desideriamo sempre inserirvi anche della musica ma spesso non siamo in grado di programmarla, neppure quella più semplice. Questa utility offre ben 20 brani musicali accompagnati da 10 disegni. Per poter ascoltare un brano è sufficiente premere il tasto corrispondente (da "1" a "P").
Le musiche ed i disegni possono essere utilizzati nei propri programmi e sono memorizzati uno ogni mille numeri di linea, partendo dalla linea 1000 per le musiche e dalla linea 30000 per i disegni. I brani sono stati realizzati con istruzioni PLAY a singola voce, escluso "L'usignolo" e "Adeste fideles".
È da notare che il carattere copyright è stato realizzato con uno sprite (linee 150-270).
Buon ascolto!

IPPICA
("IPPICA")

di E. Medvet

Vi trovate con gli amici all'ippodromo per le corse dei cavalli. Vi possono partecipare fino a 9 giocatori.
Lo scopo del gioco è scommettere sul primo cavallo arrivato (modo 1) o sui primi due (modo 2).
Dopo aver digitato il numero di giocatori si passa a puntare. Bisogna inserire il tipo di puntata (1=vincente o 2=piazzati), su quali cavalli (inserire il numero del cavallo da 1 a 6 per il modo 1 o due numeri di cavalli per il modo 2) e quanti soldi si vogliono puntare (digitare la cifra, tenendo conto della disponibilità della cassa, e poi premere ENTER).
Dopo che tutti i giocatori hanno fatto le loro scommesse il computer visualizzerà la pista di gara con i sei cavalli pronti a partire non appena viene premuto il tasto di spazio.
Durante la gara se un cavallo diventa di colore nero vuol dire che ha rotto l'andatura e che quindi non è da considerarsi più in gara. In base al risultato della corsa la cassa di ogni giocatore viene aggiornata e se questa si riduce a zero si è eliminati.
Il movimento dei cavalli è ottenuto con tre sprites di formato grande.

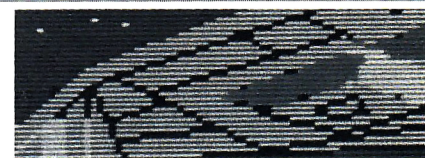
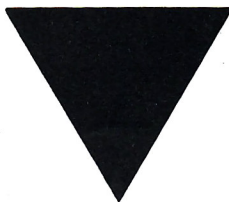
AMERICAN FOOTBALL
("FOOTBA")

di N. Paggin

Si gioca con l'ausilio della sola tastiera. Lo scopo dei giocatori è di guadagnare terreno fino ad arrivare alla cosiddetta area di TOUCH DOWN. Se un giocatore entra in quest'area come risultato di una corsa o di un passaggio, ottiene 7 punti. Al disotto delle 40 yardes dall'area di TOUCH DOWN è possibile effettuare un calcio che se riuscito fa fruttare 3 punti.

Il giocatore in attacco dispone di 4 tentativi (DOWN) per cercare di correre 10 yardes. Nel caso che egli riesca a superarle gli verranno dati altri 4 tentativi per correre altre 10 yardes. In caso contrario la palla andrà all'avversario. Il quarto tentativo è solitamente utilizzato per respingere la palla il più lontano possibile con un CALCIO DI RINVIO.

Dopo aver dato i nomi e i colori delle squadre al computer il giocatore di sinistra deve prepararsi al calcio di rinvio. Premendo il tasto di sinistra del cursore la palla partirà dal centro del campo verso sinistra. A questo punto bisogna calciare la palla premendo il tasto di spazio. Più vicino al giocatore di sinistra sarà colpita la palla e più lungo sarà il calcio di rinvio. Attenzione però a non colpire la palla troppo tardi perché altrimenti il possesso passerà all'avversario. La schermata successiva darà tutti i dati riguardanti la partita. Vengono infatti indicati la posizione sul campo, il tempo mancante alla fine del quarto

PLOTTER
("PLOT")

di M. Belardi

L'MSX offre buone qualità grafiche ma per sfruttarle appieno bisogna avere anche i mezzi giusti come questo programma assai completo e di facile uso.

Dopo la presentazione il programma chiederà nell'ordine se si vuole usare la tastiera (tasti cursore) o il joystick; che tipo di schermo usare (alta o bassa risoluzione); se si vuole lavorare un disegno nuovo, quello corrente, uno vecchio o se si vogliono cambiare le opzioni precedentemente selezionate; dopodiché si procede alla scelta del colore di fondo e si entra nel vivo del programma. La composizione del disegno si avvale di diversi oggetti. Il primo oggetto che si avrà "in mano" è la matita; gli oggetti si spostano nelle 8 direzioni e premendo il pulsante (spazio o fire) si provoca l'evento che varia a seconda dell'oggetto.

Toccando il bordo superiore compare la banda dei colori; per scegliere un colore si sposta l'oggetto su di esso e si preme il pulsante; il "corpo" dell'oggetto diventerà di quel colore e disegnerà con quello. Per far scomparire la banda dei colori basta spostare l'oggetto di lato. Toccando invece il bordo inferiore comparirà la banda degli oggetti; per uscire da questa banda la procedura è la stessa di quella dei colori; per cambiare oggetto bisogna operare nello stesso modo del cambio di colore. Ed ora vediamo le diverse possibilità del programma.

Banda dei colori: è composta da 15 rettangoli colorati più un quadrato con su scritto X/Y. Premendo il pulsante con l'attrezzo su uno dei rettangoli si "prende" quel colore. Se si seleziona invece X/Y si attiveranno o si disattiveranno (se già attivati) la coordinata X in basso a sinistra e la coordinata Y in alto a destra.

Matita: serve per disegnare a "mano libera"; premendo il pulsante disegna col colore selezionato.

Gomma: serve per cancellare. Premendo il pulsante cancella i tracciati sottostanti riportando il colore originale del fondo.

Pennello: colora le figure chiuse con lo stesso colore del pennello.

Compasso: consente di tracciare ellissi e cerchi. Premendo la prima volta viene disegnato il centro del cerchio. Poi, spostando il compasso orizzontalmente e premendo una seconda

di gioco, il possesso di palla (indicato dall'ovale sopra il nome delle squadre), il punteggio, il numero di tentativi e le yardes da correre per completare il down. Premendo la barra spaziatrice appariranno sullo schermo le

opzioni della difesa che verranno selezionate premendo il tasto corrispondente (naturalmente è opportuno che l'attaccante non veda la mossa del difensore). Toccherà quindi all'attaccante scegliere la sua mossa. Egli può scegliere tra tre diverse opzioni:

1) Corsa 2) Calcio 3) Lancio.

1) Corsa: selezionata questa opzione apparirà un altro elenco di possibilità sul tipo di corsa da effettuare. Al termine della scelta il computer giocherà l'azione mostrandone il risultato.

2) Calcio: qui vi sono due possibilità: il calcio di rinvio già descritto e il calcio da tre punti. Se scegliete quest'ultima opzione apparirà una schermata con una palla ovale in primo piano sulla sinistra e sullo sfondo la porta. Dopo qualche istante un ovale più piccolo partirà dalla sinistra dello schermo. È necessario premere ora la barra spaziatrice quando questo ovale si trova al centro del pallone (o appena prima). Naturalmente la precisione deve essere maggiore se ci si trova lontani dalla porta.

3) Lancio: vi verrà chiesto a che distanza intendete lanciare il pallone (minimo 3 yards, massimo 50 yards). Il lanciatore è l'omino col braccio alzato mentre il ricevitore è il vostro uomo alla sua destra. Per regolare la forza del lancio bisogna premere il tasto di spazio. Tanto più a lungo verrà premuto e tanto più lungo sarà il lancio (è necessario fare un certo numero di prove per trovare il tempo giusto). Se il lancio sarà corretto avanzerete di un numero di yards pari alla lunghezza del lancio. Se il lancio sarà sbagliato di poco esso sarà considerato "incompleto" e perderete solo un tentativo. Se invece l'errore sarà grosso allora la palla andrà all'avversario ("intercettata").

I tempi sono 4 di 15 minuti ciascuno. In modo assolutamente casuale comparirà talvolta un arbitro che infliggerà delle penalità.

volta viene fissato il raggio orizzontale; infine muovendo il compasso verticalmente premendo la terza volta viene fissato il raggio verticale e tracciata l'ellisse. Per tracciare un cerchio perfetto è sufficiente premere due volte il raggio orizzontale o verticale.

Riga: serve a tracciare linee in tutte le angolazioni.

Quadrato: premendo la prima volta si disegna un punto che sarà uno spigolo del quadrilatero; poi si sposta il quadrato e si preme la seconda volta.

Lettera: si sposta la lettera nel punto dove si vuole iniziare a scrivere e si preme il pulsante; a questo punto si battono sulla tastiera i caratteri, dopodiché si torna al controllo dell'oggetto premendo ENTER.

Retino: serve per colorare con puntini una figura chiusa.

Freccia: si torna al menù di scelta. Il disegno non viene salvato.

S: serve per salvare in memoria il disegno sul quale si sta lavorando senza uscire.

C: consente di riprodurre più volte anche ruotata una parte del disegno su altri punti del foglio. L'immagine da leggere non deve essere maggiore di 2500 pixel. Per eseguire la copia si punta lo spigolo del quadrilatero immaginario che racchiude la parte di disegno da copiare e si preme il pulsante, poi si marca lo spigolo opposto. Una "C" al centro del quadrato segnalerà l'avvenuta memorizzazione dell'immagine; contemporaneamente una L rovesciata delimita l'angolo superiore sinistro dell'immagine; l'angolo inferiore destro è delimitato dalla freccetta del quadrato. A questo punto si sposta il quadrato nel punto dove si vuole riprodurre il disegno e si preme il pulsante. La L rovesciata serve per le dimensioni.

Registrazione del disegno.

Il programma guida l'utente nelle varie fasi. Nella fase d'inserimento nome bisogna digitare 6 caratteri che verranno visualizzati e utilizzati come file-name.

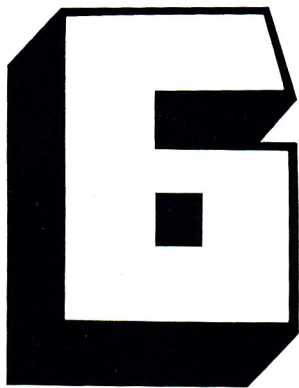
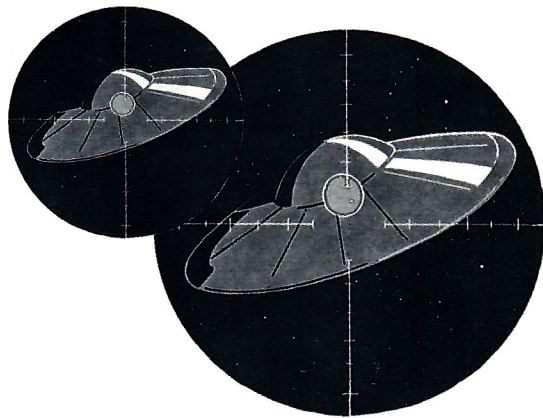
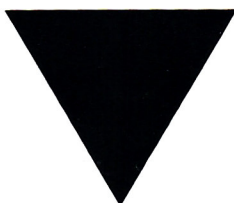
Caricamento di un disegno registrato precedentemente.

Le procedure sono identiche a quelle della registrazione; terminato il caricamento verrà visualizzato il disegno.

Utilizzazione di un disegno realizzato con PLOTTER.

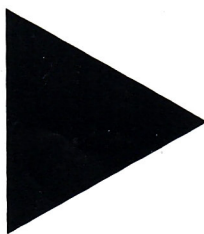
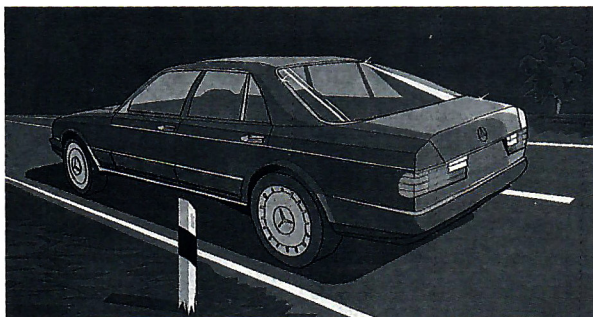
Per utilizzare un disegno in un altro programma bisogna inserire alcune linee all'inizio del programma che si deve realizzare:

```
10 CLEAR xxx,47999
20 BLOAD "xxxxxx"
30 DEFUSR=61060
40 D=USR (0)
```

La linea 10 determina l'indirizzo massimo utilizzabile da MSX-BASIC, xxx è la quantità RAM per le variabili stringa. La linea 20 carica il disegno e la routine in L/M nella RAM dalla locazione di memoria 48000 in poi; xxxxxx è il nome del disegno con cui è stato salvato. La 30 definisce l'indirizzo di partenza della routine di trasferimento mentre la linea 40 esegue il L/M. Questa istruzione può ovviamente essere inserita in qualsiasi parte del programma.

UFO
("UFO")
di F. Bonati

Nuovamente gli UFO hanno attaccato la vostra città e voi siete stati ingaggiati come pilota di un caccia da inseguimento.

Il gioco è impostato su quattro quadri diversi per quattro diverse difficoltà. Nel primo quadro un UFO appare di notte nel cielo di una città industriale: sirene, fari e l'UFO è localizzato. Subito si alza in volo il vostro caccia, veloce e munito di un'arma a raggio laser. Gli UFO da abbattere per superare il primo quadro sono 10 e devono essere colpiti prima dello scadere del tempo. Le difficoltà nel secondo quadro aumentano in quanto, oltre all'UFO, dal cielo scendono degli invaders, i quali sicuramente s'impossesseranno della città se voi non distruggete i 10 UFO in tempo utile. Potete anche respingere gli invasori urtandoli o sparandogli. Il terzo quadro si svolge fuori dall'atmosfera e dovete sempre abbattere 10 UFO, ma delle micidiali meteoriti vi ostacolano e voi dovete schivarle o colpirle. Un'astronave madre degli UFO combatte sullo sfondo. Nel quarto quadro, oltre agli asteroidi, dovete fare molta attenzione ad un terribile invader (quello color rosso, radioattivo) che fuoriesce dall'astronave madre e che, se vi colpisce, vi distrugge all'istante. A vostra volta voi potrete colpirlo; se ce l'avete fatta (bravissimi!) potete continuare con il quinto, sesto, settimo... quadro!

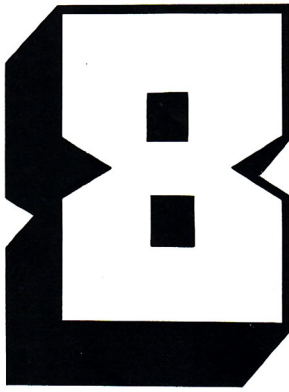
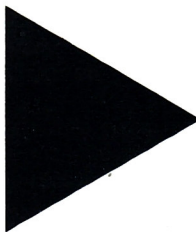
La parte di programma dalla linea 610 alla linea 730 esegue il suono.

CODICE STRADALE
("STRADA")

di G. Riccobono

Conoscete bene i cartelli stradali? Provate a confrontarvi con questo programma che vi insegnerà e vi interrogherà in modo divertente sui principali cartelli stradali.

Il menù principale presenta due opzioni: visione, interrogazione.

Premendo il tasto "V" si vedranno apparire in successione le schede ministeriali per l'esame della patente di guida. La scheda sarà visualizzata nella parte inferiore dello schermo, mentre in quella superiore vi sarà una macchinina trainante la lettera corrispondente alla risposta esatta. Per passare alla visione delle schede successive premere un tasto qualsiasi.

Premendo, invece, il tasto "P" il computer interrogherà in successione due concorrenti. Ad ogni concorrente è associata una macchinina; rossa per il primo, blu per il secondo. Nella parte bassa dello schermo apparirà una scheda ministeriale alla quale dovrà rispondere il concorrente di turno. Inizia a rispondere il giocatore della macchinina rossa. La risposta va inserita premendo il tasto della lettera a cui è associata (A, B o C). Se la risposta data è esatta la macchinina del concorrente si sposterà di un passo verso sinistra, altrimenti un'ambulanza recupererà i "feriti" e la macchinina del giocatore non avanzerà. A questo punto una nuova scheda verrà visualizzata e la risposta dovrà essere data dal secondo giocatore. Vince chi arriva per primo all'estremità sinistra dello schermo.

Dalla linea 15 alla linea 388 vi è la parte di programma che visualizza le diverse schede ministeriali.

ROTATE
("ROTATE")
di E. Dassi

Ecco un'utility capace di trasformare un vostro disegno bidimensionale in tridimensionale.

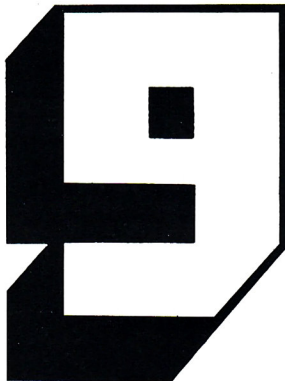
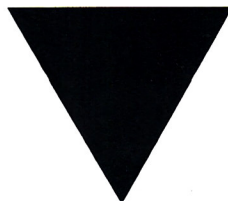
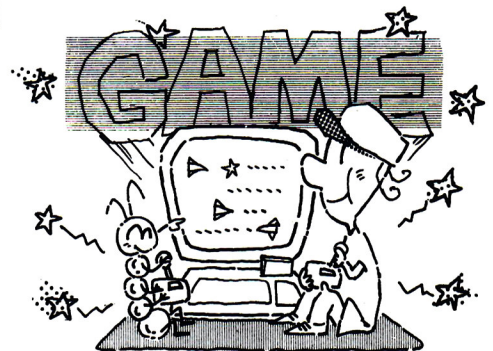
Le opzioni disponibili sono quattro e vengono selezionate con i tasti cursore su e giù. Una volta puntata la freccia sull'opzione desiderata premere spazio. Ed ora vediamo le possibilità di questo programma.

Disegnare: dopo aver inserito il codice del colore inchiostro e carta si passa a realizzare il disegno. Ora lo schermo è diviso in due parti, in quella di destra verrà realizzato il disegno. Una croce, che può essere spostata con i tasti cursore, individua la posizione di tracciatura. Premendo la barra spaziatrice si disegna una spezzata dall'ultimo punto tracciato alla posizione centrale della croce. Si possono tracciare fino a 16 linee, ma se il vostro disegno ne necessita di meno, una volta finito premere il tasto ESC (escape). A questo punto il grafico realizzato viene fatto ruotare sul proprio asse generando una figura solida. Finita l'elaborazione in alto a destra sullo schermo apparirà un cursore lampeggiante ad indicare che il nuovo disegno è completato. A questo punto premendo il tasto di spazio si ritorna al menù principale per le successive operazioni.

Stampare il disegno: questa opzione vi permette di avere una copia del disegno 3D su stampante.

Salvare il disegno: dopo aver inserito il nome del disegno (massimo 6 caratteri) predisponete il vostro registratore al salvataggio del file.

Caricare il disegno: permette di ricaricare in memoria i disegni precedentemente salvati.

Per chi volesse utilizzare i disegni in qualche programma, l'intero display file dello screen 2 viene memorizzato a partire dalla locazione 48700 fino alla locazione 60100.

SPRITES GRANDI
("SPRITE")

di T. Leddi

Uno dei punti forti dell'MSX è la possibilità di gestire facilmente da Basic gli sprite, di cui quelli grandi (formato 2x2 caratteri) sono i più utilizzati. La parte però più noiosa del loro utilizzo è la definizione del profilo, cioè della loro forma.

Ricordiamo a chi mastica per le prime volte il basic che per gli sprites è necessario in generale assegnare a ciascun punto il valore zero oppure il valore uno: è proprio in tal modo che dopo si formerà la figura che abbiamo impostato. Infatti il livello zero significa per il computer che il punto non deve essere visualizzato. Il livello uno significherà che invece desideriamo visualizzare proprio quel punto. Naturalmente i dati devono essere impostati nel programma o direttamente o convertendoli in numeri.

La grandezza degli sprite così come la scala vengono definite dall'istruzione "SCREEN modo, tipo sprite". Le coordinate di visualizzazione devono essere nell'intervallo - 32/255 l'ascissa, in - 32/191 l'ordinata.

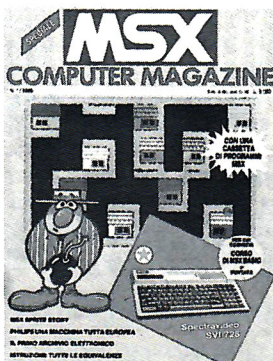
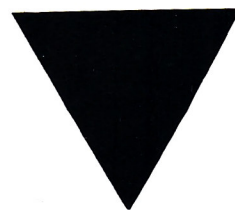
Le cose ovvero gli sprite a livello di definizione son noiose.

Ora però con questa utility il lavoro diverrà più facile.

Il programma fornisce la codifica in decimale su due colonne dello sprite disegnato.

I tasti di controllo:

tasti cursore = movimento puntatore disegno
"A" = disegna o cancella un punto in posizione cursore
"Y" = fornisce il negativo dello sprite
"T" = forma lo sprite e visualizza la codifica numerica
"C" = cambia il colore dello sprite
"F" = modifica il formato dello sprite
"X" = cancella la griglia del disegno
"M" = ritorna al menù senza perdere il disegno

La gestione della tastiera avviene nelle linee 1940-2010.

ARCHIVIO LIBRI ("ARCLIB")

di D. Montresor

Tutti noi abbiamo in casa un sacco di libri o di dischi che trattano argomenti diversi. E pure di autori diversi. Vediamo di fare un po' d'ordine aiutandoci con questo programma.

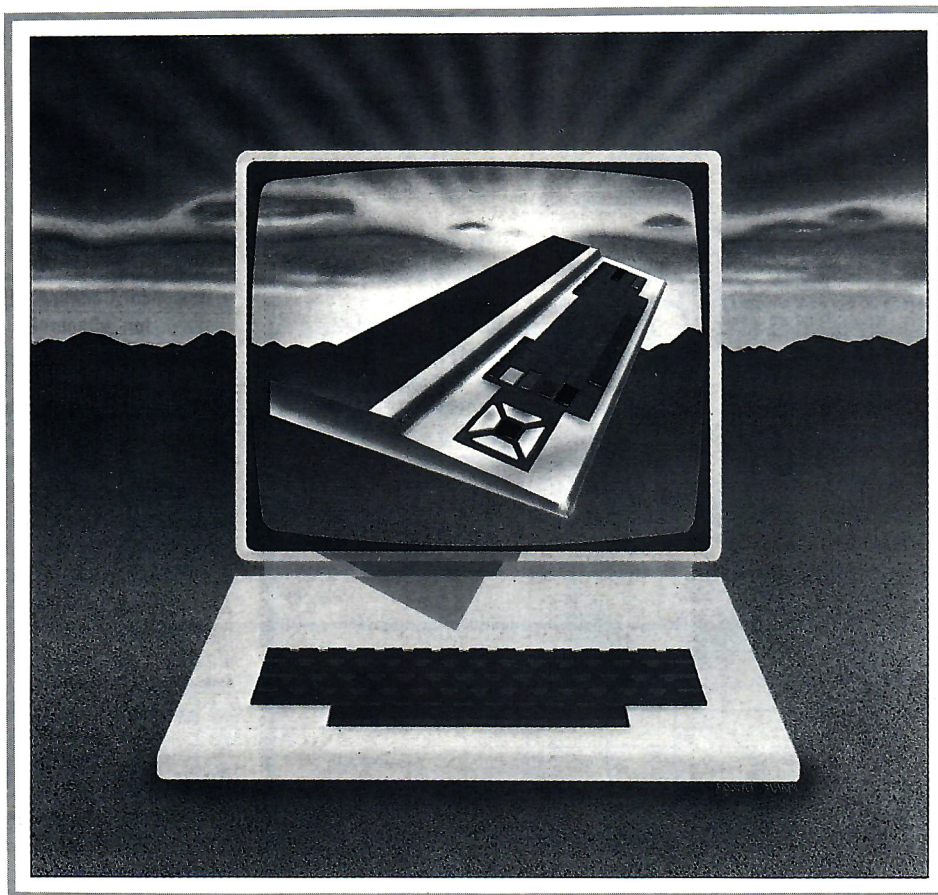
Con un MSX a 16K utente si possono memorizzare fino a 56 libri, mentre con 32K si arriva a 250.

Appena dato il RUN si deve scegliere la velocità di registrazione dei file su nastro (1 bassa = 1200 baud, 2 alta = 2400 baud). Successivamente appare il menù principale con le sue sette opzioni:

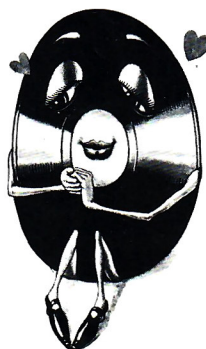
1) il sottomenù prevede quattro opzioni:
   1) fornisce il numero di schede occupate e di quelle libere.
   2) Inserzione di un libro tramite il suo numero (da —32768 a +32767 intero), titolo, autore e genere.
   3) Cancellare tutto l'archivio.
   4) Cancellare una scheda conoscendone il numero.

2) Leggere il file archivio da nastro (load).
3) Scrivere il file archivio su nastro (save).
4) Viene visualizzata tutta la libreria; premere SPACE per passare alla schermata successiva oppure SELECT per tornare al menù principale.
5) Si passa al secondo sottomenù dove è possibile effettuare una ricerca con uno qualsiasi dei campi, tranne quello numerico, scrivendo anche solo una parte di quello che si vuole cercare. Verranno trovate tutte le schede che contengono i caratteri scritti.
6) Accensione motore registratore.
7) Spegnimento motore registratore.

Per tornare al menù principale da un punto qualsiasi del programma basta premere il tasto SELECT.

Per chi volesse adattare il programma all'uso del Quick Disk o del Floppy Disk, la parte di programma per il salvataggio e il recupero file su nastro è dalla linea 4000 alla 6400.

MSX BASIC

INTRODUZIONE AL LINGUAGGIO DEI MICROCOMPUTER MSX STANDARD

(6ª PUNTATA)

di E. DASSI

Siamo giunti all'ultimo appuntamento di questo corso di MSX-Basic e per concludere tratteremo un aspetto assai affascinante dei computers: la musica.

L'MSX possiede un generatore di suoni programmabile (PSG AY-3-8910) a tre canali musicali indipendenti più un generatore di rumore.

Esistono due istruzioni per poter ottenere della musica: PLAY e SOUND. PLAY è utilizzata per suonare brani musicali ed è di facile utilizzo, SOUND serve principalmente per ottenere particolari effetti sonori (esplosioni, rumori metallici ecc.) e il suo utilizzo richiede una buona conoscenza del PSG.

L'istruzione PLAY, come detto prima, è di facile uso perché utilizza l'MML (Music Macro Language), una serie di subcomandi musicali utilizzabili similarmente a quelli del GML dell'istruzione DRAW presentata nello scorso numero. Quindi l'istruzione PLAY per poter funzionare dovrà essere seguita da almeno una stringa.

Prima d'addentrarci nella spiegazione dell'MML dobbiamo ricordare che i nomi delle note vengono espresse nella notazione anglosassone che equivale alle seguenti lettere:
DO=C; RE=D; MI=E; FA=F; SOL=G; LA=A; SI=B.

Per poter eseguire l'intera scala scriveremo:
PLAY "CDEFGAB".

Volendo invece alzare la nota di un semitono (diesis) o abbassarla di un semitono (bemolle) bisognerà posticipare la nota rispettivamente del segno più (+) o del segno meno (—); attenzione però che le combinazioni "E+", "F—", "B+" e "C—" non valgono, proprio come nella vera scala musicale.

Le note possono essere generate su ben 8 scale differenti; per selezionare l'ottava desiderata bisogna utilizzare il subcomando 'O' seguito da un numero intero compreso tra 1 e 8. Provate il seguente esempio:
PLAY "O5CDEFGAB"
PLAY "O7CDEFGAB"
le due scale sono prodotte su due ottave differenti. Se non viene specificata nessuna ottava, l'istruzione PLAY considererà l'ultima ottava definita, che per default è 4.

C'è un subcomando che permette di raggruppare tutti gli effetti appena visti: 'N'. Esso deve essere seguito da un numero intero compreso tra 0 e 96 permettendo di suonare qualsiasi nota inclusa nelle 8 ottave. Ovviamente 'N96' corrisponde alla nota più acuta mentre 'NO' non produce alcun suono fornendo invece una breve pausa.

Fino adesso abbiamo visto dei subcomandi per ottenere il suono di note ma non per controllare la loro durata. A tal fine l'MML dispone di tre subcomandi: 'T', 'L' e 'R'.

'T', seguito da un intero compreso tra 32 e 255, determina il numero di semiminime eseguite in un minuto. Il valore di default è 120. Provate le seguenti due istruzioni e ascoltate la differenza:
PLAY "T200CDEFGAB"
PLAY "T50CDEFGAB"

Ovviamente l'esecuzione del primo PLAY sarà più rapida della seconda, proprio perché il valore di 'T' è maggiore rispetto a quello della seconda istruzione.

In tutti i brani musicali la durata delle note non è mai la stessa e quindi non è sufficiente impostare il tempo d'esecuzione ma bisogna anche specificare, se necessario, il tempo di ogni nota. Il subcomando dedicato a questo scopo è 'L' che deve essere seguito da un intero compreso tra 1 e 64.

Ecco i significativi relativi ad 'L':

Schema logico dei 14 registri interni del PSG (Programmable Sound Generator) AY-3-8910. Philips courtesy.

L1= semibreve
L2= minima
L4= semiminima
L8= croma
L16= semicroma
L32=biscroma
L64= semibiscroma

Maggiore è il valore del numero seguente 'L' e minore è la durata della nota; il valore di default è 4. È chiaro che 'L3' specifica una nota da un terzo, cioè un terzo di semibreve. Inoltre è utile sapere che la durata di una nota può essere specificata immediatamente dopo la nota stessa, cioè anziché scrivere 'L16B' si può scrivere 'B16'; così come aggiungendo uno o più punti dopo una nota la sua durata viene moltiplicata per 2/3 tante volte quanti sono i punti. Per esempio scrivere 'B2' equivale a 'B3'.

Per impostare una pausa bisogna utilizzare il subcomando 'R' seguito da un numero intero compreso tra 1 e 64 il cui significato è uguale a quello già analizzato per 'L'.

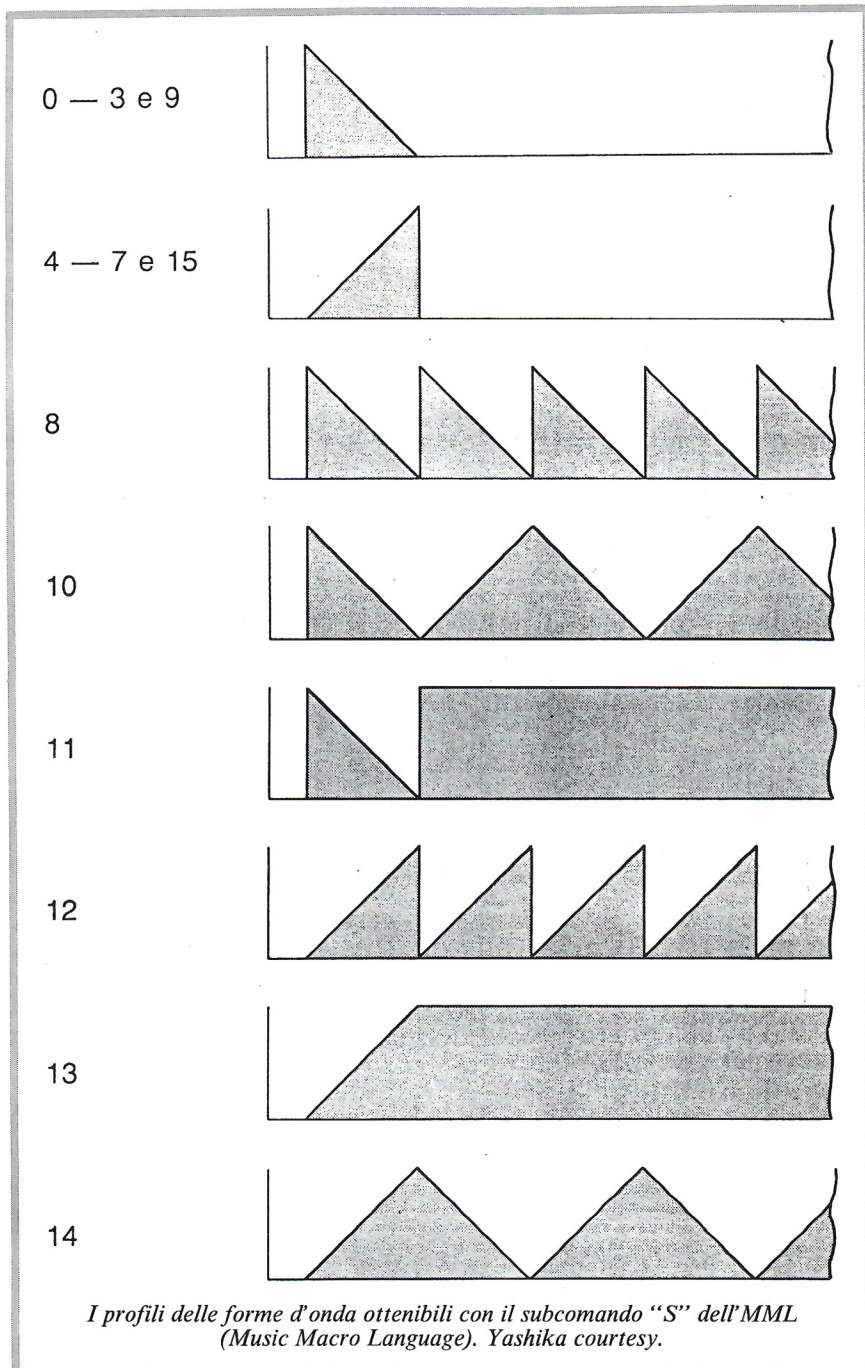
L'MML è completato da altri tre subcomandi: 'V', 'S' e 'M'.
'V' determina l'ampiezza del segnale cioè il volume e deve essere seguito da un intero compreso tra 0 e 15, il valore di default è 8. Provate le seguenti due istruzioni e, mantenendo costante il volume del proprio televisore, ascoltate la differenza:

```
PLAY "V15CDEF"
PLAY "V5CDEF"
```

Il subcomando 'S' deve essere seguito da un numero intero compreso tra 0 e 15 il quale seleziona il tipo di forma d'onda (vedi figura 1). Il subcomando 'M', invece, permette di modificare la modulazione dell'onda. 'M' deve essere seguito da un numero intero compreso tra 1 e 65535.

Provate i seguenti esempi.

```
PLAY "T200SOM1000CDE"
PLAY "S8M1000CED"
PLAY
  "S11M8000L64CDEFGAB05C"
PLAY "M300CDEFGAB"
```

Avete ascoltato questi suoni dagli effetti carini? L'MML è davvero potente, non solo per le caratteristiche musicali di cui dispone ma anche per la facilità d'uso nei propri programmi. Infatti, così come per l'istruzione DRAW, anche per il comando PLAY è possibile assegnare delle variabili alfanumeriche e numeriche; le prime inserendole nella stringa PLAY anticipate dal carattere 'X' e posticipate dal simbolo ';', le seconde inserite tra il carattere '=' e ';'.

Ecco un esempio che chiarirà quanto detto:

```
10 A$="T200"
20 DURATA=64
30 PLAY "XA$;L=DURATA,CDE"
```

L'istruzione PLAY può essere strutturata in modo tale da generare contemporaneamente tre voci indipendenti; in tal caso la sintassi dell'istruzione assumerà la seguente forma:

PLAY X1$,X2$,X3$

dove X1$,X2$ e X3$ sono le tre stringhe di subcomandi MML relative ai tre canali.

I profili delle forme d'onda ottenibili con il subcomando "S" dell'MML (Music Macro Language). Yashika courtesy.

MSX BASIC

Abbiamo detto all'inizio di questa puntata che l'utilizzo del comando SOUND è difficile perché richiede la conoscenza del PSG. In questa sede non vogliamo dare un'ampia e dettagliata descrizione dell'AY-3-8910 ma cercheremo semplicemente di conoscere un po' di più la struttura logica del processore sonoro fornendo differenti esempi applicabili in diversi programmi.

Il PSG di 14 registri di controllo di un byte ciascuno, quindi ognuno di essi può contenere un valore compreso tra 0 e 255. Vediamo ora il significato di ogni registro:

Registro 0 e 1: periodo del gener. A
Registro 2 e 3: periodo del gener. B
Registro 4 e 5: periodo del gener. C
Registro 6: periodo del rumore
Registro 7: mixer
Registro 8: volume canale A
Registro 9: volume canale B
Registro 10: volume canale C
Registro 11 e 12: periodo del generatore d'inviluppo
Registro 13: forma d'onda del generatore d'inviluppo

Vediamo come programmare la frequenza di un canale, per esempio vogliamo inserire una frequenza di 4000 Hz nel canale A (abbiamo scelto A ma potrebbe essere B o C: il metodo è lo stesso). Per prima cosa dobbiamo abilitare l'ascolto del canale A attraverso il registro del mixer resettando il bit 0, quindi scriveremo:

SOUND 7,&B11111110

Ora dobbiamo impostare il valore 4000 nei registri 0 e 1. Per far questo dobbiamo innanzitutto sapere che il contenuto del registro 1 (così come quello del registro 3 e 5) può avere un valore compreso tra 0 e 15 e tale valore deve essere moltiplicato per 256 e sommato al valore del registro 0. Ora il periodo del tono TP è dato dalla seguente formula:

$$TP = \frac{1789770}{16 \text{ x ft}}$$

dove ft è la frequenza di uscita del canale (nel nostro caso 4000 Hz). Calcolato TP dobbiamo inserire il suo valore nei registri 0 e 1 attraverso la seguente formula:

$$\frac{TP}{256} = \frac{T}{256} + CT$$

dove T è il valore da inserire nel registro 0 (o 2 o 4 a seconda se si tratta del canale B o C) e CT è il valore da inserire nel registro 1 (o 3 o 5 se si tratta del canale B o C).

Quindi con una frequenza di 4000 Hz TP è uguale a 27.96 che per approssimazione arrotondiamo a 28, mentre T e CT risultano rispettivamente 28 e 0.

Per sentire il suono, dopo aver abilitato il mixer per il canale A, scriveremo:

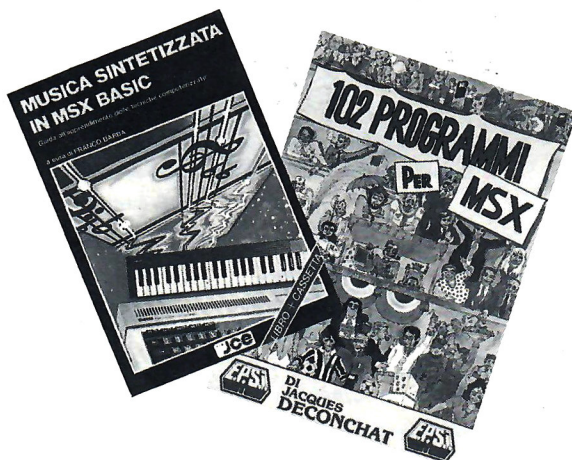
SOUND 0,28 : SOUND 1,0

Regoliamo il volume del canale A con:

SOUND 8,10

e dovremmo sentire un sibilo ininterrotto. Per fermare il fischio dobbiamo modificare i registri del PSG o con l'istruzione SOUND o più facilmente con un CTRL+STOP o BEEP.

Le operazioni sopra viste possono essere eseguite per ognuno dei tre canali. Ecco un esempio per ascoltare

LIBRI PER IMPARARE

Giunti in redazione, "Musica MSX" (di F. Barba, ed. JCE, Lire 30 mila) e "102 Programmi MSX" (di J. Deconchat, ed. Epsi, Lire 22 mila). Il primo dopo aver descritto i comandi musicali dell'MSX-Basic, passa alla descrizione del software musicale su cartridge che consente l'emissione di 8 note ed 8 voci contemporaneamente trasformando l'MSX in un vero e proprio sintetizzatore musicale professionale. La lettura di questo volume è agevolata da numerosi esempi di listati sia in MSX-Basic che in Music Bios System, tutti registrati nella cassetta allegata.

Il secondo serve ad imparare il Basic MSX senza far fatica e divertendosi.

I cento e più programmi presenti nell'opera sono suddivisi secondo 5 livelli, ognuno dei quali permette lo studio di un gruppo d'istruzioni. Il capitolo dedicato al quinto livello è il più difficile e contiene le istruzioni di lettura e scrittura nella memoria video più altro ancora.

tutti i suoni riproducibili attraverso il canale C:

```
10 SOUND 7,&B11111011
20 SOUND 10,10
30 FOR N%=0 TO 15
40 FOR M%=0 TO 255
50 SOUND 5,N% : SOUND 4,M%
60 NEXT M%,N%
70 BEEP
```

Date RUN e udirete un effetto speciale simile, inizialmente, al suono di un missile.

Il registro 6 contiene la frequenza di rumore che però è unica per tutti e tre i canali. Il programma che segue genera tutti i possibili rumori attraverso il canale B:

```
10 SOUND 7,&B11101101
20 SOUND 9,10
30 FOR N%=0 TO 31
40 SOUND 6,N%
45 FOR N=1 TO 100: NEXT N
50 NEXT N%
60 RUN
```

Il suono udito può assomigliare... ma lasciamo alla vostra immaginazione interpretare quanto ascoltato. Per interrompere il programma dare CTRL+STOP.

PER CONCLUDERE

Avete sentito quali effetti è possibile produrre con l'istruzione SOUND? Certo non abbiamo detto tutto perché lo spazio in questa sede non ce lo concede, però vogliamo dirvi che per sapere usare bene l'istruzione SOUND bisogna eseguire diversi esperimenti e a tal proposito vi consigliamo di riprendere il programma 'SOUND EXPLORER' apparso sul numero 1 di MSX COMPUTER MAGAZINE. Con esso potrete sbizzarrirvi con i registri del PSG senza dover ogni volta scrivere il comando SOUND e relativi dati.

■

CHI HA VINTO LA GRAPH BALL

La Graph Ball, prodotta e distribuita dalla Sony.

A Flavio Bonati di Monza, autore del programma "Ufo" pubblicato in questo stesso fascicolo, è stata assegnata la Graph Ball della Sony. A Flavio i complimenti della redazione per il suo bel lavoro e la cartridge con il fantastico programma per creare disegni.

Il premio era stato promesso sul fascicolo n. 3 della nostra rivista. Molti coloro che si son fatti vivi con idee e listati simpatici: a tutti grazie e ad maiora. Qualcuno chiede a noi il programma grafico: basta andare al rivenditore più vicino e chiedere del materiale Sony! Il programma, adatto a qualsiasi configurazione RAM, è venduto in cartridge e viene fornito insieme ad un particolare strumento, chiamato track-ball, che consente di disporre dell'utility.

Il track-ball è costituito da una scatola nera nella quale vi sono tre tasti comando e una sfera (di plastica bianca) che può ruotare liberamente. Muovendo quest'ultima si può controllare, in qualsiasi direzione, il tracciato del pennino sullo schermo e quindi si può disegnare qualsiasi figura. Il software, come già spiegato (cfr. numero citato), è davvero eccezionale.

ASSEMBLER

IL LINGUAGGIO MACCHINA

COME PROGRAMMARE IN LINGUAGGIO MACCHINA. I CODICI ISTRUZIONE DEL MICROPROCESSORE Z80 A

(1ª PUNTATA)

Più volte nella descrizione delle caratteristiche di un MSX si è parlato di microprocessore Z80A (la sigla A sta ad indicare la versione più veloce dello Z80) ma molti lettori si chiederanno quale funzione svolge espressioni complesse, ma come tutti i linguaggi ad alto livello anche il Basic è un programma scritto in codici numerici per Z80 che svolgono la funzione di interprete Basic. I codici istruzione dello Z80 vengono chiama-

Chip montato del microprocessore Z80, il componente più importante dell'MSX.

questo dispositivo nel microcomputer. Lo Z80 è il componente elettronico più importante dell'MSX. Ogni azione programmata viene svolta e controllata da questo integrato ed è per questo che viene considerato il cuore del computer.

Lo Z80 per poter funzionare deve essere programmato e le sue istruzioni sono una serie di codici numerici. L'MSX Basic è un linguaggio ad alto livello perché riconosce istruzioni di facile comprensione e utilizzo, nonché ti linguaggio macchina o Assembler.

In particolare l'Assembler è l'insieme dei mnemonici dei codici operativi dello Z80. Per esempio l'istruzione Assembler LD A,n è il codice mnemonico di 62, quest'ultimo interpretabile dallo Z80. Quando si programma in linguaggio macchina, che dà la massima velocità d'esecuzione di un computer, si utilizza un programma detto assemblatore che codifica automaticamente i mnemonici assembler in codici macchina risparmiando all'utente la fatica di codificare ogni istruzione.

Prima di passare alla descrizione delle istruzioni Z80 è importante conoscere un minimo di struttura interna del microprocessore per poter poi comprendere alcune istruzioni particolari.

La CPU Z80 (CPU sta ad indicare la sigla central processing unit) è un integrato a 40 piedini sui quali vengono trasmessi segnali a due stati logici: 0 e 1. Di questi 40 piedini 8 sono de-

dicati al bus dati: l'insieme dei segnali che individuano il codice dei dati o delle istruzioni da eseguire. Altri 16 piedini rappresentano il bus degli indirizzi utilizzato dalla CPU per indirizzare i dati alla memoria, infine 13 piedini rappresentano i segnali di controllo della CPU e del sistema: quei segnali dedicati al collegamento dello Z80 con altri dispositivi quali memorie, processori (per esempio il TMS 9929, per la gestione della grafica) e così via. Per quanto riguarda l'interno del microprocessore ci interessa sapere solo che vi sono 22 registri con i quali la CPU esegue differenti operazioni. Questi registri sono divisi in tre gruppi: i registri principali, i registri secondari e i registri di uso speciale. I registri principali sono 8 tutti ad 8 bit aventi i seguenti nomi: A oppure accumulatore, F o registro dei flag, B, C, D, E, H ed L. Eccetto il registro A ed F, tutti gli altri possono essere usati in coppia (BC, DE ed HL) formando così un unico registro a 16 bit. Soprattutto importante è il registro A con il quale possono essere eseguite particolari operazioni; altrettanto importante è il registro F dei flag il quale contiene differenti informazioni in seguito ad operazioni aritmetiche, di rotazione, shift, ecc.

Il gruppo dei registri secondari è identico a quello dei registri principali ma non possono essere usati contemporaneamente. Infine vi sono i registri di uso speciale comprendenti due registri ad un byte (I ed R) e quat-

ERRATA CORRIGE

LA MEMORIA FANTASMA

```
10 CLEAR 200,&HF32F
20 GOSUB 64000
30 DEFUSR=&HF330:DEFUSR1=&HF343
40 '*** DIMOSTRAZIONE ***
50 CLS
60 A%=USR(11500):PRINT "IL DATO LETTO NE
LLA LOCAZIONE 11500  ";"E':";A%
70 PRINT:PRINT"ORA ESEGUO UNA POKE 11500
,10":A%=USR1(11500):A%=USR1(10)
80 A%=USR(11500):PRINT:PRINT "IL DATO LE
TTO NELLA LOCAZIONE 11500  ";"E':";A%
90 END
64000 '*****************************
64010 '*                           *
64020 '*         POKE & PEEK       *
64030 '*    NEI 32K RAM INFERIORI  *
64040 '*            BY             *
64050 '*      ORIGINAL   SOFT.     *
64060 '*                           *
64070 '*****************************
64080 RESTORE 64100
64090 FOR N=&HF330 TO &HF372:READ A$:POK
E N,VAL("&H"+A$):NEXT:RETURN
64100 DATA 3E,FF,D3,A8,23,23,4E,23,46,0A
,36,00,2B,77,3E,F0,D3,A8,C9,3A,72,F3,CB,
7F,20,10,23,23,4E,23,46,79,32
64110 DATA 71,F3,78,CB,FF,32,72,F3,C9,CB
,BF,32,72,F3,47,3A,71,F3,4F,23,23,3E,FF,
D3,A8,7E,02,3E,F0,D3,A8,C9,00,00
```

Ci scusiamo con i lettori ma abbiamo riscontrato un errore di stampa nell'articolo "La memoria fantasma" (vedi MSX Computer Magazine n. 5). La porta descritta in questo articolo non ha il numero &HAB bensì &HA8. Inoltre il programma è stato sviluppato su computer MSX Yashika. Per differenze hardware è possibile che il programma non funzioni sul vostro computer. In tal caso inserite nel listato Basic la seguente linea:

25 POKE &HF331,n:POKE &HF367,n:POKE &HF33F,j:POKE &HF36D,j

provando il programma con n uguale a 170, 85 oppure 0 e j uguale a 160 o 80.

tro registri a due byte: IX, IY, SP e PC. Il registro I è detto anche registro del vettore interruzione ed è utilizzato per la gestione delle interruzioni (argomento che tratteremo nelle prossime puntate), il registro R (rinfresco memoria) serve per le operazioni di rinfresco delle memorie dinamiche.

I registri IX e IY vengono detti registri indice, possono essere utilizzati anche per le operazioni aritmetiche ma il loro uso principale consiste nell'indirizzamento indicizzato della memoria. Il registro PC o contatore di programma contiene l'indirizzo della locazione di memoria avente il codice operativo da interpretare; infine il registro SP, puntatore di pila, contiene l'indirizzo della prima locazione libera dello stack. Lo stack (pila) è una struttura logica di memorizzazione dei dati nella quale vale la regola LIFO (last in first out) cioè l'ultimo dato immesso sarà anche il primo ad essere prelevato. Nella pila i dati vengono memorizzati uno "sopra" all'altro per valori d'indirizzamento memoria decrescenti. Particolari istruzioni sono dedicate a questo tipo di gestione dati.

I registri dello Z80 sono 22 ma potremmo aggiungerne un altro ad un bit: il flip-flop d'interruzione IFF. Questo registro indica lo stato di abilitazione, o meno, del servizio delle interruzioni mascherabili, argomento, come già detto, che tratteremo nelle prossime puntate.

Abbiamo visto i registri, ora, prima d'iniziare la descrizione delle operazioni, è opportuno fare anche un piccolo accenno ai metodi d'indirizzamento.

Le istruzioni dello Z80 sono composte da uno a quattro bytes e possono essere suddivise in tre parti:
codice operativo — primo operando — secondo operando

Il codice operativo indica l'operazione richiesta dall'istruzione mentre gli operandi indicano le unità coinvolte nell'operazione (registri, celle di memoria, buffer d'ingresso-uscita dati). Per esempio, nell'istruzione LD

A,10 il codice operativo è LD, il primo operando è A ed il secondo è 10. A seconda delle modalità con le quali vengono indicati gli operandi si ottengono i vari metodi d'indirizzamento che tratteremo di volta in volta insieme alla presentazione di nuove istruzioni.

Incominciamo quindi con le istruzioni di caricamento. I registri coinvolti in queste operazioni sono sette (A,B,C,D,E,H ed L) più quelli indici per alcune istruzioni. Una delle istruzioni di maggior comprensione è LD r,n (LD sta per LoaD) dove r è uno dei sette registri ed n un valore numerico compreso tra 0 e 255. Il significato di questa istruzione è molto semplice: carica nel registro r il valore n; quindi dopo l'esecuzione del comando LD B,2 avremo nel registro B il valore 2. Questo tipo d'istruzione appartiene al metodo d'indirizzamento immediato perché il valore del secondo operando è direttamente contenuto nell'istruzione.

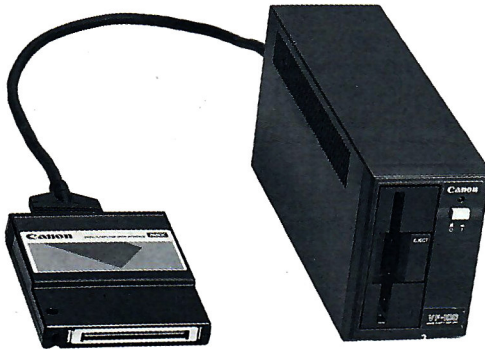
Un registro può contenere anche il valore di un altro registro con l'istruzione generale LD d,s dove d è il registro destinazione ed s il registro sorgente. Tutte le combinazioni sono ammesse ed è per questo che esistono ben 63 diverse istruzioni LD che coinvolgono solo registri singoli comprese LD A,A oppure LD E,E apparentemente inutili. Abbiamo visto l'indirizzamento immediato esteso. Le istruzioni appartenenti a questo gruppo usano come operando destinazione una coppia di registri a 8 bit e come sorgente un valore numerico a 16 bit, come descritto dalla seguente forma generale: LD cr,nn dove cr è la coppia di registri (BC, DE, HL nonché IX, IY ed SP) e nn è il valore numerico compreso tra 0 e 65535. Da quest'ultima istruzione di LoaD si può già osservare come lo Z80 offra una vasta scelta di operazioni a seconda dei tipi di dati da trattare.

Fino adesso abbiamo visto caricamenti immediati, spesso invece bisogna trattare numerosi dati memorizzati in RAM e caricarli a turno nei registri. In assembler vi sono differenti istruzioni che svolgono questa funzione. Per esempio, volendo caricare l'accumulatore con il valore contenuto nella cella d'indirizzo 25000, utilizzeremo l'istruzione LD A,(25000). È da notare che le parentesi indicano che il dato sorgente non è 25000 ma quello contenuto nella cella d'indirizzo 25000. Questa istruzione utilizza un metodo d'indirizzamento indiretto. Gli altri registri godono di un caricamento del tipo appena visto se uniti in coppia nella seguente forma sintattica:

LD cr, (nn)

dove cr è BC,DE,HL,IX,IY e SP, mentre nn è una locazione di memoria d'indirizzo compreso tra 0 e 65535 e non il valore sorgente.

In questa puntata abbiamo descritto numerosi e importanti concetti della programmazione Assembler Z80, la prossima volta continueremo con la descrizione delle istruzioni e incominceremo con qualche esempio pratico su computer MSX.

CANON, QUASI UN MEGABYTE!

Già in vendita nei migliori negozi italiani (ahimé, costa poco meno di un milione!) il nuovo floppy disk drive da 3,5" Canon VF 100, che può immagazzinare dati fino ad 1 megabyte. Qui di seguito, alcune delle caratteristiche più rilevanti.

INTERFACCIA
PROGRAMMI INTERNI: Capacità 16K Bytes
Routines Basic Entrata/Uscita
Routines Basic MSX-DOS
MSX-Disk Basic
Programmi utilitari
CONSUMO DI CORRENTE: +5 V 300 mA massimo
LETTORE
SUPPORTO DATI: Bifacciale, doppia densità, doppia pista, Minidisco da 3,5 pollici
CAPACITÀ DI MEMORIA: 1M byte (non formattato)
720K bytes (formattato)
Settore: 512 bytes
Pista: 9 settori/pista
Numero di piste: 80 piste x 2
DENSITÀ DI REGISTRAZIONE: 8717 BPI
METODO DI REGISTRAZIONE: MFM (Modified Frequency Modulation)
VELOCITÀ DI ROTAZIONE: 300 RPM
VELOCITÀ DI TRASMISSIONE DEI DATI: 250K bits/secondo
TEMPO MEDIO DI ATTESA DELLA ROTAZIONE: 100 ms
TEMPO MEDIO DI ACCESSO: 95 ms
TEMPO DI SPOSTAMENTO DELLA TESTINA: 6 ms
TEMPO DI POSIZIONAMENTO DELLA TESTINA: 15 ms
TEMPO DI ATTESA DELLA ROTAZIONE: 100 ms
ALIMENTAZIONE E CONSUMO CORRENTE: 220 V, 50/60 Hz, massimo 25 W
TEMPERATURA DI FUNZIONAMENTO: 5 — 35° C
DIMENSIONI ESTERNE: 31 mm x 130 mm x 252,2 mm
PESO: 2,7 kg.

UNA MAGLIETTA IN REGALO!

a chi si abbona a

MSX COMPUTER MAGAZINE

sei magnifiche cassette di programmi di gioco e di utilità, sempre più belle e ricche!

il prezzo dell'abbonamento (Lire 50 mila) è bloccato per sei numeri e non ti verranno quindi richiesti aumenti (già subito intanto risparmi 4 mila lire)!

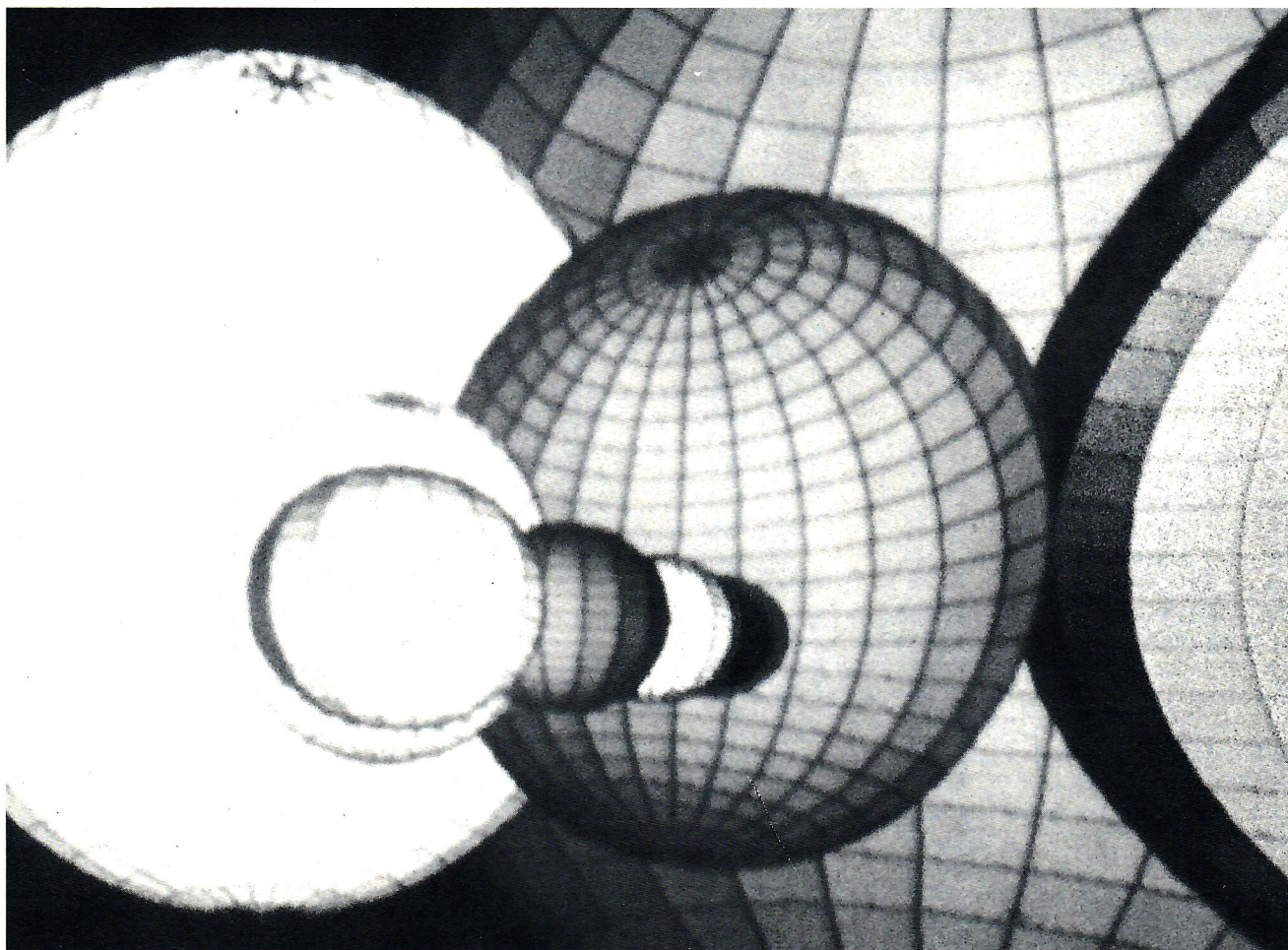
avrai subito, direttamente a casa, un'elegante maglietta (realizzata con le riviste consorelle Elettronica 2000 e Load'n'Run) assolutamente gratis!

ABBONATI OGGI STESSO

**Basta inviare un vaglia ordinario (quello rosa, da richiedere in un qualunque ufficio postale) di lire 50 mila.
Indica esattamente da quale fascicolo desideri l'abbonamento ed i tuoi dati chiari e precisi. Indirizza a MSX Computer Magazine, C.so Vitt. Emanuele, 15 - 20122 Milano.**

SCROLL WINDOW

UNA ROUTINE PER ESEGUIRE PROGRAMMI CON EFFETTI GRAFICI DAVVERO SPECIALI

di BEN NOYA

EIDOS COURTESY

Un buon programma deve anche prevedere una buona impostazione grafica la quale va attentamente studiata a seconda del problema spesso proprio particolare da risolvere.

Prima di passare a descrivere la routine grafica che verrà trattata in questo articolo, chiariremo il concetto di window, indispensabile per comprendere la funzionalità della routine in esame: vedi più avanti il listato.

Per window (finestra) s'intende una porzione del video nella quale è possibile scrivere e variare quanto contenuto senza disturbare la zona circo-

stante. Molti programmi applicati a macchine aventi la possibilità di gestire le finestre, utilizzano questa risorsa software per creare speciali effetti di impaginazione rendendo il programma più facile nell'uso e maggiormente comprensibile.

La routine è stata realizzata in linguaggio macchina ed opera solo nel modo testo 1 cioè SCREEN 0. La scelta del far operare la routine solo in screen 0 risiede nel fatto che è solo in questo modo grafico che si ha la massima quantità di caratteri sul video. Sappiamo che nel modo testo 1 si possono avere fino a 40 colonne per 24 righe, abbiamo quindi 960 locazioni di memoria (40×24) contenenti i valori ASCII dei caratteri rappresentati sullo schermo; tali locazioni vanno dall'indirizzo 0 all'indirizzo 959 della VRAM (la memoria dedicata al video). Per capire come è disposta tale parte di memoria provate il seguente programmino:

```
10 SCREEN 0
20 WIDTH 40
30 FOR N=0 TO 959
40 VPOKE N,65
50 NEXT N
```

Date il RUN e vedrete rapidamente riempire le linee di video da sinistra verso destra con il carattere 'A', il cui codice ASCII è appunto 65. La routine, dopo aver definito le dimensioni della finestra, esegue uno scroll dal basso verso l'alto. Per eseguire uno scroll verso l'alto bisogna copiare i contenuti delle linee inferiori in quelle

ECCO IL PROGRAMMA

IN BASIC

```
10 CLEAR 200,&HF000
20 GOSUB 64000
30 DEFUSR0=&HF001
40 X=&HF0A0:Y=&HF0A1
50 YLN=&HF0A4
60 SCROLL=&HF0A5
70 SCREEN 0: WIDTH40:KEYOFF
80 FOR R1=1000 TO 1 STEP-300
90 LOCATE 0,0:PRINT "FINESTRA DI COORDIN
ATE 0,0":PRINT "DI LUNGHEZZA 26":PRINT"E
 DI ALTEZZA 4":PRINT"SCROLL MODO 1"
100 POKE X,0:POKE Y,0:POKE XLN,26:POKE Y
LN,4:POKE SCROLL,1
110 FOR R=1 TO 5:GOSUB 210:A=USR(0):NEXT
 R
120 NEXT R1
130 LOCATE 0,0:PRINT "FINESTRA DI COORDI
NATE 0,0":PRINT "DI LUNGHEZZA 26":PRINT"
E DI ALTEZZA 4":PRINT"SCROLL MODO 1"
140 POKE X,10:POKE Y,10:POKE XLN,16:POKE
 YLN,5:POKE SCROLL,2
150 LOCATE 10,10:PRINT "FINESTRA DI":LOC
ATE 10,11:PRINT"COORDINATE 10,10":LOCATE
 10,12:PRINT "DI LUNGHEZZA 16":LOCATE 10
,13:PRINT"E DI ALTEZZA 5":LOCATE 10,14:P
RINT"SCROLL MODO 2"
160 FOR R1=1000 TO 1 STEP -300
170 FOR R=1 TO 6:GOSUB 210:A=USR(0):NEXT
 R,R1
180 FOR R1=10 TO 20:FOR R=1 TO 5:GOSUB 2
10:A=USR(0):NEXT R,R1
190 GOTO 30
200 REM CICLO DI PAUSA
210 FOR N=1 TO R1:NEXT N:RETURN
64000 '*****************************
64010 '*                           *
64020 '*   SUBR. SCROLL WINDOW     *
64030 '*                           *
64040 '*            BY             *
64050 '*                           *
64060 '*      ORIGINAL SOFT.       *
64070 '*                           *
64080 '*****************************
64090 RESTORE 64110
64100 FOR N%=&HF001 TO &HF0A5:READ A$:PO
KE N%,VAL("&H"+A$):NEXT N%
64110 DATA 3A,A1,F0,CD,6D,F0,3A,A3,F0,06
,00,4F,11,CE,F0,C5,CD,59,00,C1,3A,A1,F0,
3C,32,A2,F0,CD,6D,F0,11,A6,F0
64120 DATA C5,F5,CD,59,00,F1,C1,3D,CD,6D
,F0,54,5D,21,A6,F0,C5,CD,5C,00,C1,3A,A1,
F0,21,A4,F0,86,3D,21,A2,F0,BE
64130 DATA 28,05,3A,A2,F0,18,CE,3E,01,21
,A5,F0,BE,20,0C,3A,A2,F0,CD,6D,F0,3E,00,
CD,56,00,C9,3A,A2,F0,CD,6D,F0
64140 DATA 54,5D,21,CE,F0,CD,5C,00,C9,6F
,26,00,CB,25,CB,14,CB,25,CB,14,CB,25,CB,
14,CB,25,CB,14,CB,25,CB,14,E5,26,00,6F,C
B,25,CB,14,CB,25,CB,14,CB
64150 DATA 25,CB,14,D1,19,F5,3A,A0,F0,5F
,16,00,F1,19,C9,00,00,00,28,18,01
64160 RETURN
```

superiori. Uno scroll interno ad una window è più complesso perché bisogna tener conto delle dimensioni della window stessa che possono andare da quella di un carattere a quelle dell'intero video (40 × 24 caratteri).

La subroutine si colloca a partire dalla locazione 61441 ed è lunga 244 bytes; è possibile caricarla tramite il listato Basic oppure, se disponete di un programma assemblatore, tramite il listato Assembler.

Per poter funzionare bisogna definire 5 variabili:

X= coordinata orizzontale dell'angolo in alto a sinistra della finestra (valore minimo 0 valore massimo 40)

Y= coordinata verticale dell'angolo in alto a sinistra della finestra (valore minimo 0 valore massimo 24)

XLN= lunghezza orizzontale della finestra (valore minimo 1 valore massimo 40)

YLN=lunghezza verticale della finestra (valore minimo 1 valore massimo 24)

MODO=definisce il tipo di scroll (se 1 esegue uno scroll normale se 2 fa rientrare il testo, uscente dalla parte alta, nella zona bassa)

Ecco le locazioni di memoria relative ai contenuti delle variabili:

X= &HF0A0 (61600)

Y= &HF0A1 (61601)

XLN= &HF0A3 (61603)

YLN= &HF0A4 (61604)

MODO= &HF0A5 (61605)

Supponiamo di voler definire una finestra di lunghezza 10, di altezza 12 e di coordinate dell'angolo superiore sinistro 11,8 con scroll normale. Il programma di gestione sarà il seguente:

10 POKE &HF0A0,11: REM setta la coordinata X della window
20 POKE &HF0A1,8: REM setta la coordinata Y della window
30 POKE &HF0A3,10: REM setta la lunghezza della window
40 POKE &HF0A4,12: REM setta l'altezza della window
50 POKE &HF0A5,1: REM setta il tipo di scroll
. . . .
. . . .
resto del programma
. . . .
. . . .
100 A=USR(0)
. . . .
. . . .
resto del programma
. . . .

IN ASSEMBLER

```
F001         0010         ORG  0F001H
F001 3AA1F0  0020         LD   A.(Y)
F004 CD6DF0  0030         CALL CALC
F007 3AA3F0  0040         LD   A.(XLN)
F00A 0600    0050         LD   B.0
F00C 4F      0060         LD   C.A
F00D 11A8F0  0070         LD   DE.BUF2
F010 C5      0080         PUSH BC
F011 CD5900  0090         CALL 0059H
F014 C1      0100         POP  BC
F015 3AA1F0  0110         LD   A.(Y)
F018 3C      0120 CONT    INC  A
F019 32A2F0  0130         LD   (Y1).A
F01C CD6DF0  0140         CALL CALC
F01F 11A6F0  0150         LD   DE.BUF1
F022 C5      0160         PUSH BC
F023 F5      0170         PUSH AF
F024 CD5900  0180         CALL 0059H
F027 F1      0190         POP  AF
F028 C1      0200         POP  BC
F029 3D      0210         DEC  A
F02A CD6DF0  0220         CALL CALC
F02D 54      0230         LD   D.H
F02E 5D      0240         LD   E.L
F02F 21A6F0  0250         LD   HL.BUF1
F032 C5      0260         PUSH BC
F033 CD5C00  0270         CALL 005CH
F036 C1      0280         POP  BC
F037 3AA1F0  0290         LD   A.(Y)
F03A 21A4F0  0300         LD   HL.YLN
F03D 86      0310         ADD  (HL)
F03E 3D      0320         DEC  A
F03F 21A2F0  0330         LD   HL.Y1
F042 BE      0340         CP   (HL)
F043 2805    0350         JR   Z.FINE
F045 3AA2F0  0360         LD   A.(Y1)
F048 18CE    0370         JR   CONT
F04A 3E01    0380 FINE    LD   A.1
F04C 21A5F0  0390         LD   HL.MODO
F04F BE      0400         CP   (HL)
F050 200C    0410         JR   NZ.FINE2
F052 3AA2F0  0420         LD   A.(Y1)
F055 CD6DF0  0430         CALL CALC
F058 3E00    0440         LD   A.0
F05A CD5600  0450         CALL 0056H
F05D C9      0460         RET
F05E 3AA2F0  0470 FINE2   LD   A.(Y1)
F061 CD6DF0  0480         CALL CALC
F064 54      0490         LD   D.H
F065 5D      0500         LD   E.L
F066 21A8F0  0510         LD   HL.BUF2
F069 CD5C00  0520         CALL 005CH
F06C C9      0530         RET
F06D 6F      0540 CALC    LD   L.A
F06E 2600    0550         LD   H.0
F070 CB25    0560         SLA  L
F072 CB14    0570         RL   H
F074 CB25    0580         SLA  L
F076 CB14    0590         RL   H
F078 CB25    0600         SLA  L
F07A CB14    0610         RL   H
F07C CB25    0620         SLA  L
F07E CB14    0630         RL   H
F080 CB25    0640         SLA  L
F082 CB14    0650         RL   H
F084 E5      0660         PUSH HL
F085 2600    0670         LD   H.0
F087 6F      0680         LD   L.A
F088 CB25    0690         SLA  L
F08A CB14    0700         RL   H
F08C CB25    0710         SLA  L
F08E CB14    0720         RL   H
F090 CB25    0730         SLA  L
F092 CB14    0740         RL   H
F094 D1      0750         POP  DE
F095 19      0760         ADD  HL.DE
F096 F5      0770         PUSH AF
F097 3AA0F0  0780         LD   A.(X)
F09A 5F      0790         LD   E.A
F09B 1600    0800         LD   D.0
F09D F1      0810         POP  AF
F09E 19      0820         ADD  HL.DE
F09F C9      0830         RET
F0A0 00      0840 X       DEFB 0
F0A1 00      0850 Y       DEFB 0
F0A2 00      0860 Y1      DEFB 0
F0A3 28      0870 XLN     DEFB 40
F0A4 18      0880 YLN     DEFB 24
F0A5 01      0890 MODO    DEFB 1
F0A6 A6F0    0900 BUF1    DEFW 0F0A6H
F0A8 CEF0    0910 BUF2    DEFW 0F0CEH
             0920         END
```

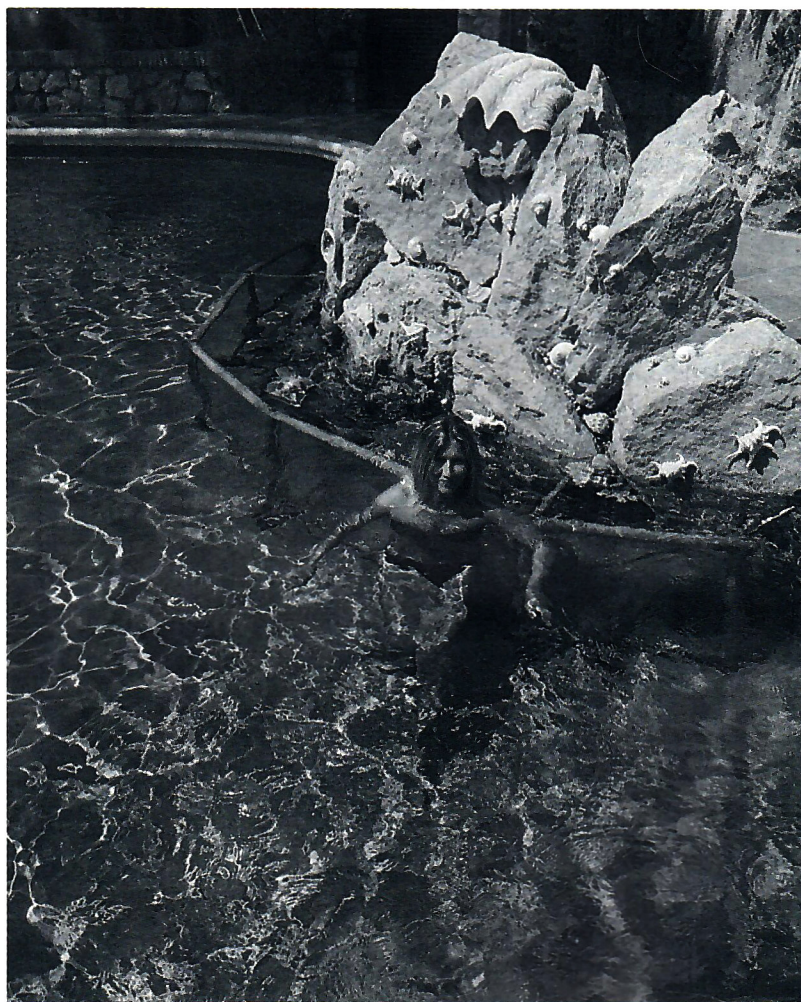
La linea 100, che per esempio è stata collocata in quella posizione, contiene la chiamata alla routine in linguaggio macchina, quindi tale istruzione sarà preceduta dal comando:

DEFUSR=61441

Per comprendere maggiormente come opera e come usare tale routine vi consigliamo di caricare il programma Basic. Una volta ammirati gli straordinari effetti di scorrimento parziale del video date il break (CTRL+STOP) e analizzate il listato. Ricordiamo che è possibile lavorare con più window contemporaneamente, l'importante è ricordarsi di aggiornare le variabili.

Ed ora commentiamo brevemente il listato Assembler. Il programma utilizza tre chiamate in ROM. La CALL 0059H trasferisce un blocco di dati dalla VRAM alla memoria utente; i dati d'ingresso sono: il registro BC, contenente la lunghezza del blocco da trasferire, il registro DE, contenente l'indirizzo della memoria utente, e il registro HL, contenente l'indirizzo della VRAM. La seconda CALL in ROM è la chiamata all'indirizzo 005CH. I registri implicati in questa subroutine sono gli stessi della precedente con la sola differenza che HL contiene l'indirizzo della memoria utente e DE l'indirizzo della VRAM; infatti questa routine trasferisce un blocco di dati dalla memoria utente e DE l'indirizzo della VRAM; infatti questa routine trasferisce un blocco di dati dalla memoria utente alla VRAM. Infine vi è la CALL 0056H che riempie, con un dato costante contenuto nel registro A, un blocco di VRAM, la cui lunghezza è indicata nel registro BC, a partire dall'indirizzo contenuto nel registro HL. La routine di scroll opera nel seguente modo: dopo aver calcolato (subroutine CALC) l'indirizzo dell'angolo in alto a sinistra della window, vengono trasferiti i dati della prima riga nel BUF2 (start FOCEH). Dopodiché vengono ricopiati, tante volte quante sono le righe della window meno una, i dati della file inferiori in quelle superiori ottenendo l'effetto di scroll. Alla fine, prima di restituire il controllo al Basic, verifica se deve eseguire uno scroll normale o con rientro, nella zona inferiore della window, del testo. Nel primo caso utilizzando la CALL 0056H si cancella l'ultima riga della finestra, nel secondo caso si riempie l'ultima fila con i dati della prima riga precedentemente salvati in BUF2.

TEMA
UNA RAGAZZA AL MARE

SVOLGIMENTO
UN CENTOMILA

Hai un computer e sai disegnare sul video. Prova a svolgere il tema proposto: una ragazza al mare, o in piscina. Con contorno magari di ombrelloni coloratissimi, del blu dell'acqua, di bikini o di monokini di moda, e di tutto quant'altro vuoi. Inviaci lo screen. Se è bello lo utilizziamo noi per la copertina (facciamo noi la foto a meno che tu non voglia già inviare una diapositiva a colori) e ti paghiamo centomila lire. Naturalmente il tuo nome viene citato come autore del disegno. La cassetta o la foto deve essere inviata in redazione (MSX Computer Magazine, Vitt. Emanuele 15, 20122 Milano), la modella può rimanere con voi (ci contentiamo dell'immagine). Il materiale inviato non potrà essere restituito se non accompagnato da indirizzo e affrancatura (visto che costi postali?!). Coraggio: le centomila son pronte per un disegno graficomputerizzato che si faccia guardare!

MSX BANK

FINO AD UN MILIONE PER UN PROGRAMMA TUO, CHE SIA ORIGINALE

Se hai programmi originali, che siano proprio pensati e fatti da te, di giochi e utilità, mandaceli in visione. Quelli pubblicati verranno compensati con almeno 200 mila lire l'uno. S'intende che se realizzerai qualcosa di davvero favoloso potrai guadagnare addirittura 1 milione di lire! Registra un solo programma per ogni cassetta, completo di istruzioni scritte, ed invialo a MSX Computer Magazine, c.so Vittorio Emanuele 15, 20122 Milano.

MODI

LA RIDEFINIZIONE DEI CARATTERI

di L. PASI

I computer MSX dispongono di quattro modi di funzionamento per quanto riguarda la visualizzazione di dati; in linguaggio Msx-Basic il tipo di funzionamento è selezionato con la variabilė X del seguente comando:

SCREEN X,Y,Z,W,J

in cui le altre variabili hanno il seguente significato:
Y = dimensione dello sprite (8×8 o 16×16), ingrandito o meno;
Z = attivazione del BEEP alla pressione di un tasto;
W = velocità di trasmissione o ricezione dati;
J = selezione tipo di stampante (Msx oppure no).

Ciò che interessa a noi di questo comando sono le prime due variabili X e Y i cui valori e significati sono riportati in tabella 1. In ogni caso l'omissione di una variabile comporta il mantenimento dello stato precedente (attenzione però a non omettere le opportune virgole).

Il passaggio da uno SCREEN ad un altro è regolato da alcune variabili chiamate BASE che forniscono il primo indirizzo di alcune tabelle di memoria situate all'interno della RAM video; di esse ce ne sono cinque per ogni SCREEN per un totale di 20 (da 0 a 19) non tutte usate. Guardando con attenzione la tabella 2 le possiamo catalogare come segue:
1) tabella con i numeri dei caratteri;
2) tabella colori dei caratteri;
3) tabella con i dati dei caratteri;
4) tabella informazioni degli sprite;
5) tabella con i dati degli sprite.

Esaminiamole ora insieme:
1) TABELLE CON I NUMERI DEI CARATTERI.
La zona di memoria video che comincia con l'indirizzo specificato dalla base corrispondente contiene dei numeri che in ogni caso sono compresi tra 0 e 255 ed il loro significato rispecchia i caratteri (CHR$) di cui ogni manuale riporta l'elenco. Se per esempio troviamo un byte contenete il valore 65 decimale vuol dire che quel byte sta memorizzando il carattere "A"; infatti CHR$(65)=A.

Un'altra importante caratteristica di questa zona è che i caratteri in essa contenuti sono riportati sullo schermo a cominciare dall'alto a sinistra; ciò vale a dire che se i primi 40 byte con-

TABELLA 1

Valore di X	Modo di funzionamento	
0	Scrittura 40x24 caratteri	(SCREEN 0)
1	Scrittura 32x24 caratteri	(SCREEN 1)
2	Grafico ad alta risoluzione	(SCREEN 2)
3	A colori	(SCREEN 3)

Valore di Y	Tipo di sprite
0	8x8 non ingranditi
1	8x8 ingranditi
2	16x16 non ingranditi
3	16x16 ingranditi

TABELLA 2

SCREEN	BASE	valore	SIGNIFICATO
	0	0	inizio tabella dei numeri dei caratteri
	1	—	non usato
0	2	2048	inizio tabella dei dati dei caratteri
	3	—	non usato
	4	—	non usato
	5	6144	inizio tabella dei numeri dei caratteri
	6	8192	inizio tabella dei colori dei caratteri
1	7	0	inizio tabella dei dati dei caratteri
	8	6912	inizio tabella delle informazioni sprite
	9	14336	inizio tabella dei dati degli sprite
	10	6144	inizio tabella dei numeri dei caratteri
	11	8192	inizio tabella dei colori dei caratteri
2	12	0	inizio tabella dei dati dei caratteri
	13	6912	inizio tabella delle informazioni sprite
	14	14336	inizio tabella dei dati degli sprite
	15	2048	inizio tabella dei numeri dei caratteri
	16	—	non usato
3	17	0	inizio tabella dei dati dei caratteri
	18	6912	inizio tabella delle informazioni sprite
	19	14336	inizio tabella dei dati degli sprite

tengono il 66, stando in SCREEN 0, avremo la prima riga in alto riempita di lettere "B", oppure se fossimo in SCREEN 1 la prima riga e otto caratteri della seconda. In pratica questa tabella occupa per SCREEN 0, 40×24 caratteri ovvero 960 byte (nella VRAM essendo la BASE(1)=0, si trova da 0 a 959); per SCREEN 1 di 32×24 caratteri vale 768 (ed essendo BASE(5)=6144, si trova da 6144 a 6911).

Molto utile per verificare tutto ciò è l'istruzione VPOKE che usiamo nei seguenti esempi:

```
NEW

10 SCREEN 0:VPOKE 0,65:
   VPOKE 959,66
```

dopo aver dato il RUN nel primo quadretto in alto a sinistra vedremo la lettera "A" e nell'ultimo in basso a destra la lettera "B";

```
NEW

10 SCREEN 1: VPOKE 6144,67:
   VPOKE 6911,68
```

dopo il RUN al posto di "A" e "B" troveremo "C" e "D".

Per quanto riguarda SCREEN 2 la tabella non ha le stesse funzioni come negli altri modi di visualizzazione; comunque il discorso lo riprenderemo più avanti.

3) TABELLA DEI DATI DEI CARATTERI.

Questa zona di memoria video, che inizia con la base corrispondente (vedi tabella 2), contiene la definizione dei caratteri come se fossero degli sprite. Essendo ogni carattere definito da otto pixels (cioè otto bit) per otto linee (otto byte) la zona è quindi lunga 8 byte × 256 caratteri (CHR$) per un totale di 2048 byte (da 0 a 2047). Di otto in otto byte troviamo quindi la definizione di un carattere, ed essendo la memoria di tipo volatile ad accesso casuale (RAM) possiamo a piacere inserire delle nuove definizioni. Per esempio: per modificare il carattere "a" ricaviamo dai manuali che corrisponde al CHR$(97) per cui dobbiamo modificare otto byte a cominciare dalla locazione di memoria:

(97×8)+BASE(n)

in cui BASE(n) è l'inizio di memoria della tabella con i dati dei caratteri ricavate dalla tabella 2.

Gli otto nuovi dati li ricaviamo da una griglia di 8×8 pixels con la rappresentazione del nuovo carattere, che è poi il metodo usato per la costruzione degli sprites.

Vediamo ora un esempio pratico:

```
NEW
5 BS = BASE (7)
10 SCREEN 1 : FOR X = (97*8)
   +BS TO (97*8+BS+7)
20 READ Y:VPOKE X,Y
30 NEXT X
40 DATA255,129,129,129,129,129,
   129,255
```

Con questo programmino sostituiamo il carattere "a" con un quadrato ricavato dalla linea 40 di dati; in riga 10 si esegue un FOR-NEXT dal valore in cui comincia la definizione del carattere sommato al valore della BASE, per otto volte; la linea 20 si occupa di leggere i nuovi dati e di sistemarli in memoria. Se ora premiamo il tasto rappresentante il carattere "a" ne uscirà un quadrato.

MODI

Il programma di esempio va bene anche per SCREEN 0 ma con l'accorgimento di modificare la linea 5 in: BS=BASE(2) e la 10 con: SCREEN 0:... Bisogna fare attenzione inoltre che per visualizzare le 40 colonne, il carattere verrà preso non più come 8×8 pixel, ma 6×8: mancheranno dunque le due colonne verticali di pixels a destra. Sapendo ciò la linea 40 di dati la faremo diventare:

```
40 DATA253,132,132,132,132,132,
   132,253
```

e le due colonne a destra del carattere le lasceremo vuote.

Per quanto riguarda SCREEN 2 la tabella con i dati dei caratteri contiene direttamente ciò che è poi visualizzato nello schema; quest'ultimo è

sempre diviso in 32×24 ideali caratteri, solo che ognuno è tenuto in memoria non con il proprio numero nella tabella dei numeri, ma come gruppo di 8 byte di 8 pixels nella tabella dei dati dei caratteri.

2) TABELLE DEI COLORI DEI CARATTERI

Questa non viene utilizzata in SCREEN 0, infatti solo con il comando COLOR possiamo variare i colori dei caratteri e dello sfondo. In SCREEN 1 esiste invece la tabella che permette di modificare i colori; essa è riferita ai caratteri indifferentemente dalla posizione che occupano nello schermo. Purtroppo c'è una limitazione: i colori possono essere dichiarati solo per gruppi di otto caratteri alla volta, per un totale di 256(CHR$)/8(gruppi)=32, da 0 a 31. Quindi da CHR$(0) a CHR$(7) possiamo dare un solo colore per lo sfondo ed un solo colore per l'inchiostro con:

```
VPOKE (BASE(6)+n),X
```

con n=0 e dove i quattro bit alti di X contengono le informazioni dell'inchiostro ed i quattro bit bassi quelle dello sfondo. Da CHR$(8) a CHR$(15) un'altra definizione con:

```
VPOKE (BASE(6)+n),X
```

con n=1 questa volta. Da CHR$(16) a CHR$(23) sempre con la stessa formula ma con n=3; e così via fino ad un massimo di 31. I numeri di riferimento dei colori sono riportati sui manuali.

Vediamo ora un esempio per SCREEN 1:

```
10 SCREEN 1: COLOR
   15,1,1:CLS
20 FOR X=0 TO 31
30 READ Y:VPOKE (BASE(6)
   +X),Y
40 NEXT X
50 FOR V=1 TO 2:FOR W=30
   TO 255:PRINTCHR$(W);
   :NEXT W: NEXT V
60 DATA31,32,34,35,36,37,38,39,
   40,41,42,43,44,45,46
70 DATA101,102,103,104,105,
   106,107,108
80 DATA109,110,111,112,113,114,
   115,116
```

In riga 10 prepariamo lo schermo per SCREEN 1 e diamo dei colori di fondo. Da riga 20 a riga 40 inseriamo i dati nella memoria video; in riga 50 visualizziamo i caratteri dal numero 30 al 255 (in CHR$) che avranno già i nuovi colori. I dati per i colori inseriti nelle righe 60, 70 e 80 sono così ricavati: il primo (31) è equivalente all'espressione (1×16)+15=31 in cui 1 (=colore nero) moltiplicato per 16 (i quattro bit alti) servirà per il colore del carattere e 15 (=colore bianco) sarà per lo sfondo (i 4 bit bassi); e così per gli altri dati delle righe.

Dopo aver dato il RUN lo schermo si riempirà di simpatici caratteri colorati.

Per SCREEN 2 vale quanto già detto per la tabella con i dati dei caratteri; nella zona di memoria che comincia col valore di BASE (11) e per 6144 byte risiedono le informazioni dei colori per ogni singola barra di 8 pixels orizzontali che, lette contemporaneamente a quelle della tabella con i dati dei caratteri, vengono immediatamente trasferite sullo schermo dal VDP. È necessario dare entrambi i dati (pixels e colori) per una corretta visualizzazione.

Ed eccoci ad un esempio per SCREEN 2:

```
10 SCREEN 2:COLOR
   15,1,1:CLS
20 FOR X=0 TO 7
30 READ Y:VPOKE (BASE(11)
   +X),Y
40 NEXT X
50 DATA133,133,133,122,122,220,
   220,220:REM dati per i colori
60 FOR X=0 TO 7
70 READ Y:VPOKE
   (BASE(12)+X),Y
80 NEXT X
90 DATA15,15,15,15,15,15,15,15:
   REM dati per i pixels
100 GOTO 100
```

I NUOVI DISEGNI

Con questo esempio riempiamo i primi otto byte della tabella dei dati (righe 20-50) e i primi otto della tabella dei colori (righe 60-90); ciò comporta il riempimento del primo carattere virtuale posto in alto a sinistra dello schermo. Inserendo i dati della riga 90 in una griglia di 8×8 pixels vediamo che si crea un carattere la cui parte destra è settata, l'altra invece no (ciò per distinguere meglio i colori dei pixels settati dallo sfondo); per il calcolo dei dati di riga 50 dei colori caratteri/sfondo vale quanto già detto per i colori in SCREEN 1: modificandoli (entro un campo da 0 a 255) cambierà l'aspetto del carattere.

FAST SOFT

TRE LISTATI VELOCISSIMI PER IL TUO COMPUTER

di P. GAZZARRI

Non vogliamo proporvi chilometrici listati da ricopiare, ma solamente alcuni brevi software dalle caratteristiche interessanti, sia dal punto di vista dell'utilità, che da quello delle potenziali qualità di impiego in altri campi.

Dunque i programmi in esame devono essere considerati come esempi, per darvi l'opportunità di comprendere meglio alcune particolarità del Basic MSX.

L'estrema semplicità dei listati vi dovrà anche consentire di utilizzare i programmi, o parte di essi, per la creazione di altri e nuovi lavori.

MSX TIME SYSTEM

La variabile TIME dell'MSX memorizza il trascorrere del tempo. Può contenere un numero compreso tra zero e 65535, che viene incrementato automaticamente dal sistema di un'unità ogni cinquantesimo di secondo.

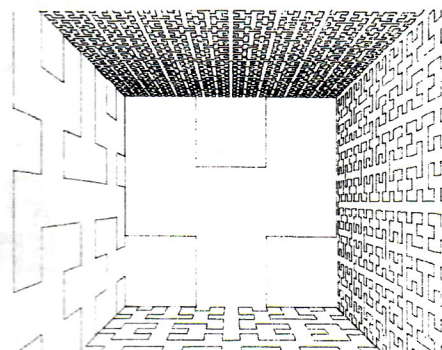
Il programma sfrutta, questa variabile per comandare un loop in cui, ogni secondo, vengono disegnate sullo schermo (in SCREEN 2) le sei cifre del quadrante, digitale dell'orologio.

L'istruzione grafica utilizzata è DRAW, in correlazione della variabile alfanumerica A$() che contiene gli spostamenti della linea sullo schermo. L'indice della variabile rappresenta il numero da tracciare. All'inizio il programma richiede l'ora, i minuti e i secondi per poter partire con il cronometraggio.

MSX TIME SYSTEM

```
10 COLOR1,11,10:SCREEN0:WIDTH36:OPEN"GRP
:"AS1:DIMA$(9)
20 A$(0)="C1U20R20D40L20U20C11R20"
30 A$(1)="C11U20R20C1D40C11L20U20R20"
40 A$(2)="C11U20C1R20D20C11D20C1L20U20R2
0"
50 A$(3)="C11U20C1R20D40L20C11U20C1R20"
60 A$(4)="C1U20C11R20C1D40C11L20U20C1R20
"
70 A$(5)="C1U20R20C11D20C1D20L20C11U20C1
R20"
80 A$(6)="C1U20R20C11D20C1D20L20U20R20"
90 A$(7)="C11U20C1R20D40C11L20U20R20"
100 A$(8)="C1U20R20D40L20U20R20"
110 A$(9)="C1U20R20D40L20C11U20C1R20"
120 H=0:TIME=0
130 FORT=0TO12:FORR=0TO5:FORE=0TO9:FORW=
0TO5:FORQ=0TO9
140 IFH=0THEN250
150 A=Q:N=200:GOSUB240
160 A=W:N=170:GOSUB240
170 A=E:N=130:GOSUB240
180 A=R:N=100:GOSUB240
190 A=T-INT(T/10)*10:N=60:GOSUB240
200 A=INT(T/10):N=30:GOSUB240
210 IFTIME<50THEN210
220 TIME=0:BEEP
230 NEXTQ,W,E,R,T:GOTO 130
240 PRESET(N,90):DRAW"XA$(A);":RETURN
250 LOCATE0,3:PRINT"Inserisci i valori:"
260 LOCATE0,5:INPUT"ORA ";T
270 IFT<0ORT>12THEN260
```

```
280 LOCATE0,7:INPUT"MINUTI ";RE
290 IFRE<0ORRE>59THEN280
300 R=INT(RE/10):E=RE-R*10
310 LOCATE0,9:INPUT"SECONDI ";WQ
320 IFWQ<0ORWQ>59THEN310
330 W=INT(WQ/10):Q=WQ-W*10
340 H=1:SCREEN2
350 PRESET(46,50):PRINT#1,"ORE":PRESET(1
16,50):PRINT#1,"MIN":PRESET(186,50):PRIN
T#1,"SEC"
360 PSET(160,100):PSET(160,80):PSET(90,1
00):PSET(90,80)
370 PRESET(88,10):COLOR6:PRINT#1,"Ora es
atta":PRESET(65,160):COLOR4:PRINT#1,"MSX
 Time Sistem"
380 COLOR1:GOTO 210
```

PROGRAMMI

DIVISIONE IN SILLABE

```
    200 KEYOFF:SCREEN0:WIDTH37:COLOR 15
,4:POKE64683!,255
210 S=1:PRINTCHR$(31)
220 INPUT"VOCABOLO ";A$:PRINT
230 IFA$="0"THENEND
240 IFS>LEN(A$)THENPRINTCHR$(29);" ":GOT
O 210
250 X$=MID$(A$,S,1):GOSUB460
260 IFA=0THEN430
270 X$=MID$(A$,S+1,1):GOSUB460
280 IFA=0THEN360
290 IFX$="I"THEN320
300 IFMID$(A$,S,1)="I"ORMID$(A$,S,1)="U"
THEN430
310 GOTO 440
320 X$=MID$(A$,S+2,1):GOSUB460
330 IFS>1THENIFMID$(A$,S-1,2)="QU"ANDA=-
1THENPRINTMID$(A$,S,2);:S=S+2:GOTO240
340 IFA=-1THEN440
350 GOTO 430
360 IFS+2>LEN(A$)THEN450
370 X$=MID$(A$,S+2,1):GOSUB460
380 IFA=-1THEN440
390 IFMID$(A$,S+1,1)=MID$(A$,S+2,1)THEN4
50
400 A1$=MID$(A$,S+1,1):IFA1$="S"ORA1$="G
"THEN440
410 A2$=MID$(A$,S+2,1):IFA2$="R"ORA2$="L
"ORA2$="H"THEN440
420 GOTO 450
430 PRINTMID$(A$,S,1);:S=S+1:GOTO 240
440 PRINTMID$(A$,S,1);"-";:S=S+1:GOTO 24
```

DIVISIONE IN SILLABE

L'applicazione degli Home-Computers, nell'analisi di testi spesso si restringe unicamente nell'uso dei Word Processors, per ovvie ragioni di memoria.

Infatti una completa analisi lessicografica (controllo di tutte le parole di un testo) in un piccolo calcolatore è praticamente impossibile per la minima capacità RAM e i lenti tempi di elaborazione di un linguaggio come il nostro Basic.

Ma i problemi maggiori sono dati proprio dalle regole grammaticali, che troppo spesso non riescono a contenere tutte le eccezioni, e costringono l'elaboratore a confrontare caso per caso. Un lampante esempio sono i plurali italiani che non sempre seguono schemi fissi, oppure gli articoli determinativi e indeterminativi da mettere prima di una parola.

Infatti, per un essere umano, è una cosa molto istintiva la creazione e l'elaborazione di una frase, mentre per un computer è un fatto certamente difficile.

Il software che vi proponiamo è l'eccezione che conferma la regola: un raro esempio di come la grammatica può rivelarsi logica quanto un algoritmo.

Il programma permette la divisione in sillabe di qualsiasi parola della lingua italiana. Può rivelarsi utile se siete un giovane studente oppure un esperto programmatore con il progetto di un sofisticato Word Processor.

I compiti del software sono quelli di scomporre il vocabolo contenuto in

```
0
450 PRINTMID$(A$,S,2);"-";:S=S+2:GOTO 24
0
460 A=X$="A"ORX$="E"ORX$="I"ORX$="O"ORX$
="U":RETURN
```

GENERATORE LABIRINTI

```
5 DIMA(4):A(1)=2:A(2)=-2:DIMB(4):B(3)=2:
B(4)=-2
10 KEYOFF:SCREEN1:WIDTH29
20 CLS:INPUT"Quante colonne (dispari) ";
N
30 IFINT(N/2)=N/2THEN20
35 IFN<3ORN>29THEN20
40 CLS
50 DIMM(N,22):M(1,1)=5
60 FORF=1TO21:FORG=1TON:PRINT"█";:NEXT
61 IFN<>29THENPRINT
62 NEXT
70 A=1:B=1:LOCATE1,1:PRINT" "
80 J=INT(RND(-TIME)*4)+1:X=J
83 IFA+A(J)<0THEN100
84 IFA+A(J)>N-2THEN100
85 IFB+B(J)<0THEN100
86 IFB+B(J)>20THEN100
90 IFM(A+A(J),B+B(J))=0THENM(A+A(J),B+B(
J))=J:M(A+A(J)/2,B+B(J)/2)=10:LOCATEA+A(
J)/2,B+B(J)/2:PRINT" ":LOCATEA+A(J),B+B(
J):PRINT" ":A=A+A(J):B=B+B(J):GOTO 80
100 J=J-1:J=J+1ANDJ<3:J=J+1:IFJ<>XTHEN83
110 J=M(A,B):IFJ<5THENA=A-A(J):B=B-B(J):
GOTO 80
120 CLEAR
```

A$ seguendo le regole più comuni della lingua italiana, rispettando anche i casi "particolari" come, ad esempio, quello di andare "a capo" sempre prima di una S, oppure quando si trovano due o più consonanti vicine, solo una (la prima) deve far parte della sillaba che precede.

CON I
CARATTERI SPECIALI

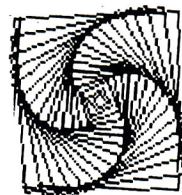
Il listato utilizza alcuni caratteri speciali del sistema MSX, i caratteri CHR$(31) e CHR$(29) corrispondono rispettivamente agli spostamenti del cursore, ovvero della posizione di stampa, verso il basso e verso sinistra.

La subroutine alla linea 460 utilizza un particolare tipo di operazione logica sulle stringhe. In pratica la variabile A conterrà il valore —1 solo se in X$ è posta una vocale, in caso contrario conterrà zero.

Per uscire dal programma inserite uno zero all'INPUT.

GENERATORE DI LABIRINTI

Prepara sullo schermo in modo testo 2 (SCREEN 1) un labirinto casuale la cui larghezza è importante dall'utente all'inizio del programma all'INPUT "Quante colonne ?". Il numero di colonne deve essere naturalmente dispari, maggiore di 3 e minore di 29.

Nelle variabili A() e B() è memorizzato lo spostamento del cursore nelle quattro direzioni. Durante la

creazione di un labirinto, l'indice di queste variabili è assunto da J che contiene un numero casuale compreso tra 1 e 4.

La variabile M() contiene la mappa del labirinto. Essa memorizza negli incroci del dedalo la cifra J per indicare al programma quale strada è stata presa dall'ultimo corridoio disegnato.

Alla linea 120 è stato inserito un CLEAR per pulire la zona destinata alle variabili. Infatti tutte le variabili impostate fino a questo momento non serviranno al programma che seguirà.

Naturalmente il CLEAR può essere evitato se si vuole mantenere la mappa del labirinto in M() oltre che sullo schermo. La variabile conterrà un numero positivo in corrispondenza di un corridoio sullo schermo, conterrà zero in corrispondenza di un muro.

Ⓐ Ⓡ Ⓣ

MSX GRAPHICS

QUALI STRUMENTI PER LA PREPARAZIONE DI PAGINE VIDEO DI ELEVATA QUALITÀ

di F. TAGLIABUE

Disegnare con il computer non è una novità. Da tempo ci vengono proposti diversi metodi. Consideriamone alcuni, prendendo in esame pregi e difetti.

Gli elaboratori MSX offrono delle notevoli possibilità grafiche attivabili direttamente da Basic. Questa prerogativa è molto interessante per poter visualizzare i dati sotto forma di istogrammi, diagrammi a torta, grafici, ma non è certo la funzionalità che permette di fare disegni al computer, così come si può fare su di un foglio di carta. Per realizzare pagine grafiche con aspirazioni "artistiche", al fine di abbellire il proprio software, o per creare giochi ed adventure con una accattivante successione di immagini, occorre far uso di strumenti software o hardware/software creati da operatori che usano le logiche degli MSX con la stessa facilità con cui adopereremmo una scatola di pastelli.

Il primo passo per lasciarsi coinvolgere del desiderio di esprimere le doti artistiche personali sullo schermo di un monitor o sul tv color consiste nell'acquistare un pacchetto software (disponibile in cassetta), come il T-Painter, (Leoni Informatica). Con questo genere di programma si cominciano ad ottenere le prime soddisfazioni; i tasti cursore o l'aventuale joystick sono le magiche penne colorate che trasformano i bit in un disegno. Questo tipo di software è semplice da usare. Mette a disposizione un menù di lavoro per selezionare l'uso di strumenti soft che possiamo assimilare ad un righello o ad un compasso. Tirare linee, tracciare circon-

A SCUOLA DI GRAPHICS

Disegnare con il computer è, per molti, solo un gioco o un hobby; per altri è diventata professione. Diversi giovani, provenienti dalle più disparate esperienze scolastiche, si sono iscritti a scuole di computer con corsi orientati verso la grafica ottenendo dei risultati brillantissimi con interessanti sbocchi lavorativi. Fra queste scuole vi segnaliamo la Eidos, che organizza corsi di computer graphic a vari livelli. Nelle aule destinate ai corsi sono disponibili ottimi elaboratori a sistemi CAD (computer aid design) operativi su grandi sistemi. Per informazioni sui corsi scrivete a Eidos, via Fontana 16, Milano.

Una videata realizzata con Mouse e Cheese.

Fra i software grafici, il T-Painter è sicuramente uno dei migliori. È stato realizzato dalla Leoni Informatica di Milano. I tasti cursore o il joystick pilotano le scelte dal menù che appare su di un lato dello schermo. Diverse funzioni permettono di ottenere disegni geometrici ed a mano libera di buona qualità.

ferenze diventa facile; riempire le sagome di colori è solo un fatto di gusto personale; una tavolozza permette di scegliere fra i colori disponibili, poi si posiziona il cursore sull'area da colorare ed in un battibaleno l'immagine si ravviva di brillanti tinte.

Il software coinvolge "i disegnatori da tastiera", e il virus della Computer Graphics è difficile da guarire. Si possono ridurre i continui bisogni solo con qualche periodico prelievo dal portafoglio per attrezzarsi...

Fra i prodotti i Mouse, le Light Pen e le Graphic Table.

Il Mouse, corredato dal Cheese (software per il pratico "topolino" che sostituisce il joystick), pone, senza dubbio, rimedio alle prime febbri intense da computer graphic. La mano non sposta più levette e, tantomeno, le dita battono sui tasti cursore; finalmente il palmo della mano impugna il Mouse, lo scatolino a sfera che scorre sul piano del tavolo come la punta della matita su di un foglio. La tecnica di lavoro è piacevole; i risultati che si ottengono con le utility grafiche contenute nel pacchetto software Cheese sono molto valide.

Quando le ansie da computer graphic si fanno più intense, nasce però un problema che non è superabile nemmeno da Mouse: il bisogno del rapporto diretto fra la mano e l'immagine.

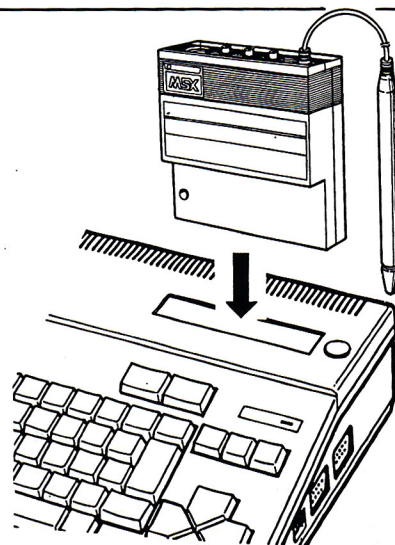
Fino alle soluzioni già considerate, questo rapporto è indiretto: la mano si muove ed un cursore traccia per noi l'immagine sullo schermo.

Con una penna ottica o una tavo-

Il Mouse, strumento molto diffuso anche fra i personal computer per uso professionale, è costituito da uno scatolino in cui è inserita una sfera che, ruotando quando il Mouse scorre su di un piano, fornisce al computer dei dati abbinabili alla posizione di un eventuale cursore sullo schermo.
Per disegnare, il rapporto fra Mouse e schermo è gestito dal Cheese: il software studiato appositamente per computer graphic.

BY A. COOK

LA PENNA OTTICA

La penna ottica è un dispositivo che permette di fornire delle indicazioni al computer in funzione del posizionamento del trasduttore (penna) sull'area dello schermo.
Fra i prodotti di questo genere per MSX vi segnaliamo quello della Sanyo, noto con la sigla MPL-001. È una penna con trasduttore e interfaccia da innestare nello slot del computer. L'unica differenza che si introduce rispetto al normale uso dell'MSX, a livello delle connessioni elettriche, consiste in un diverso posizionamento delle terminazioni video ed audio, come illustra chiaramente (con un disegno) il manuale della Light Pen Unit. Le immagini preparate con la penna ottica possono essere riportate su stampante o salvate su cassetta per eventuali richiami da Basic. Sulle pagine video, tramite comandi da menù, si possono introdurre anche testi posizionati in funzione delle disposizioni date dalla penna.
L'uso della light pen comporta la disponibilità dei comandi in Expander Basic dedicati alla grafica. La sintassi dei comandi è: Call Bline, Call Lline, Call Spaint e Call Spset.

letta grafica il rapporto con l'immagine è completamente diverso. Per quanto riguarda la penna ottica abbiamo un legame immediato punta della penna/schermo.

Nel caso della tavoletta grafica lo schermo diventa secondario: è solo la destinazione ultima dell'immagine.

La tavoletta grafica è molto facile da usare; permette di separare il momento del concepimento del disegno dall'istante in cui la forma grafica si concretizza sullo schermo. Questo è senz'altro un vantaggio se non avete il tratto sicuro, potete verificare la vostra idea su un pezzo di carta, prima di avvalervi del tocco artistico del computer. Ci sono difetti? Sì, le dimensioni della tavoletta. Tutte le tavolette grafiche per MSX oggi in commercio, o che lo saranno presto, sono piccole. Per questo motivo manca un legame dimensionale tra il disegno tracciato a matita ed appoggiato sul piano della tavoletta e l'immagine reale che si forma sul video.

Con la penna ottica il problema del rapporto dimensionale è completamente superato, lo spostamento della penna sul video corrisponde fedelmente alle linee tracciate un pixel dopo l'altro. Disegnare diventa elementare.

In conclusione: in primis può senz'altro bastare un buon software come il T-Painter; se siete entusiasti del lavoro fatto e desiderate mezzi espressivi più potenti e capaci di sbalordire gli amici, prendete in considerazione la possibilità di intervenire sulle vostre finanze acquistando una penna ottica. Se, invece, volete solo abbellire le pagine grafiche del vostro software con immagini più piacevoli, il Mouse è il meglio. Per la tavoletta grafica occhio alle dimensioni e alla risoluzione del piano della tavoletta rispetto al video.

Dunque si può cominciare. Non lasciatevi prendere dall'ansia di comperare. Provate prima il prodotto!